2006-12

Development of an artillery accuracy model

Fann, Chee Meng.

Monterey California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

DEVELOPMENT OF AN ARTILLERY ACCURACY MODEL

by

Chee Meng Fann

December 2006

Thesis Advisor: Morris Driels

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE		Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE December 2006	3. REPORT TYPE AND DATES COVERED Master's Thesis

4. TITLE AND SUBTITLE Development of an Artillery Accuracy Model	5. FUNDING NUMBERS
6. AUTHOR(S) Chee Meng Fann	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	12b. DISTRIBUTION CODE A

13. ABSTRACT (maximum 200 words)

This thesis explains the methodologies that predict the trajectory and accuracy of an unguided, indirect-fire launched projectile in predicted fire. The trajectory is the path that a projectile travels to the impact point, while the accuracy is the measurement of the deviation of the impact point from the target. In addition, this thesis describes, the methodology for calculating the various factors such as drag and drift in the trajectory calculation. A three degree of freedom model will be compared to a five degree of freedom model. With an accurate trajectory prediction, it is possible to calculate the delivery accuracy in a predicted fire, which does not have cumulative error corrections associated with the registration or adjusted fire. The delivery accuracies that are considered in this thesis are; 1) Mean Point of Impact (MPI) that are related to aiming errors and 2) Precision errors that are related to the dispersion caused by ballistics effect. Finally, the trajectory and accuracy estimates are compared with NATO Armament Ballistics Kernel (NATO) and Joint Weapons Accuracy Model (JWAM) respectively, and the differences are of the order of 4 percent.

14. SUBJECT TERMS Indirect-fire, Trajectory Model, Accuracy Model, Predicted Fire, Precision Error, Mean Point of Impact Error	15. NUMBER OF PAGES 113
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500 Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

DEVELOPMENT OF AN ARTILLERY ACCURACY MODEL

Chee Meng Fann
Civilian, Singapore Technologies Kinetics Limited
B. Eng, Nanyang Technological University, 1998

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ENGINEERING SCIENCE
(MECHANICAL ENGINEERING)

from the

NAVAL POSTGRADUATE SCHOOL
DECEMBER 2006

Author: Chee Meng Fann

Approved by: Professor Morris Driels
Thesis Advisor

Professor Anthony Healey
Chairman, Department of Mechanical and Astronautical Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

This thesis explains the methodologies that predict the trajectory and accuracy of an unguided, indirect-fire launched projectile in predicted fire. The trajectory is the path that a projectile travels to the impact point, while the accuracy is the measurement of the deviation of the impact point from the target. In addition, this thesis describes, the methodology for calculating the various factors such as drag and drift in the trajectory calculation. A three degree of freedom model will be compared to a five degree of freedom model. With an accurate trajectory prediction, it is possible to calculate the delivery accuracy in a predicted fire, which does not have cumulative error corrections associated with the registration or adjusted fire. The delivery accuracies that are considered in this thesis are; 1) Mean Point of Impact (MPI) that are related to aiming errors and 2) Precision errors that are related to the dispersion caused by ballistics effect. Finally, the trajectory and accuracy estimates are compared with NATO Armament Ballistics Kernel (NATO) and Joint Weapons Accuracy Model (JWAM) respectively, and the differences are of the order of 4 percent.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF ABBREVIATIONS AND ACRONYMS

a	Acceleration of projectile
a_o	Constant for deflection error
a_1	Constant for deflection error
a_D	Deflection acceleration
$a_{D,p}$	Deflection acceleration due to projectile
$a_{D,w}$	Deflection acceleration due to wind
a_x	Horizontal acceleration of the projectile
a_y	Vertical acceleration of the projectile
dt	Time step
g	Gravitational constant
l	Length of projectile in caliber
m	Mass of projectile
n	Length of gun per turn of rifling
t	Time
t_f	Time of flight
v	Velocity of projectile
v_o	Muzzle Velocity
v_{ox}	Horizontal component of the muzzle velocity
v_{oy}	Vertical component of the muzzle velocity
v_r	Relative velocity of the projectile
v_s	Speed of Sound
v_x	Horizontal component of the projectile’s velocity

v_y	Vertical component of the projectile's velocity
w	Wind speed
x_f	Distance of impact point in range
z	Distance in deflection

A_{base}	Cross-sectional area of the projectile
A_{wet}	Wetted area of the projectile
BATES	Battlefield Artillery Target Engagement System
C_D	Total aerodynamic drag coefficient
$C_{D(\text{wave})}$	Wave drag coefficient
$C_{D(\text{friction})}$	Friction drag coefficient
$C_{D(\text{Base})}$	Base drag coefficient
CG	Center of gravity
CP	Center of pressure
D	Drag
DOF	Degree of Freedom
F_D	Drag force
$F_{D,D}$	Drag force in deflection
$F_{D,R}$	Drag force in range
GPS	Global Positioning Systems
GTL	Gun target line
IACO	International Civil Aviation Organization
INS	Inertial Navigation Systems
JWAM	Joint Weapon Accuracy Model

L	Lift
LOS	Line of sight
M	Mach number
M_m	Magnus moment
MCS	Modular charge system
MPI	Mean Point of Impact
MRLS	Multiple Rocket Launcher System
NATO	North Atlantic Treaty Organization
NABK	NATO Artillery Ballistic Kernel
PE	Precision error
QE	Quadrant elevation
R	Gas constant
S_{ref}	Reference area
T	Thrust
TAM	Thesis Accuracy Model
X	Distance in range
Y	Height
Z	Total drift
Z_w	Drift due to wind
Z_p	Drift due to projectile

γ	Adiabatic index of air
φ	Angle between the projectile axis and trajectory
θ	Angle of the elevation with horizontal

θ_o	Quadrant elevation
$\Delta\theta$	Incremental angle in quadrant elevation
ρ	Density of air
$\sigma_{AIM\text{-}EL}$	Aiming in quadrant elevation standard deviation
$\sigma_{AIM\text{-}AZ}$	Aiming in azimuth standard deviation
$\sigma_{CHART\text{-}X}$	Chart accuracy standard deviation in range
$\sigma_{CHART\text{-}Z}$	Chart accuracy standard deviation in deflection
σ_{Drag}	Form factor
σ_{LIFT}	Lift factor
$\sigma_{LOC\text{-}X}$	Location accuracy standard deviation in range
$\sigma_{LOC\text{-}Z}$	Location accuracy standard deviation in deflection
$\sigma_{MPI,X}$	MPI error in range
$\sigma_{MPI,Z}$	MPI error in deflection
σ_{PX}	Precision error in range
σ_{PZ}	Precision error in deflection
σ_{θ}	Quadrant elevation standard deviation
σ_{ρ}	Density standard deviation
σ_{T}	Temperature standard deviation
σ_{v}	Velocity standard deviation
σ_{W}	Wind standard deviation
$\sigma_{X,v}$	Error in range due to standard deviation in muzzle velocity
$\sigma_{X,\theta}$	Error in range due to standard deviation in quadrant elevation
Λ	Coriolis Acceleration

ACKNOWLEDGMENTS

I thank my advisor, Professor Morris Driels, for his continuous support in this thesis. He was always patient and guided me throughout this thesis progress. In addition, he taught me how to express my ideas in this report. With his guidance and advice, the objective of the thesis was accomplished.

I also thank Mr Teo Chew Kwee, Mr Teo Ee Tiong, Mr Tan Kheng Huek and Mr Leonard Lye from Singapore Technologies Kinetics Limited for their continuous support.

During the course of this thesis, I received valuable insights and help from the United States Army Armament Research, Development and Engineering Center. Special thanks to James A. Matt and Andre J. Sowa for their insights and advice.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. OBJECTIVES

The main objective of this thesis is to formulate a simple methodology that can be implemented using computing software to predict the trajectory and accuracy in predicted fire for an unguided, indirect, artillery fire. In addition, this thesis discusses the contributing factors that affect both trajectory and accuracy.

In order to achieve the objective, a trajectory model is needed to predict the nominal trajectory. Thus a simple model using three Degree of Freedom (DOF) is implemented using *Matlab*. With the nominal trajectory known, the accuracy of the impact point can then be calculated using partials derivatives (also commonly known as unit effects) and error budgets of the factors that affect the accuracies.

It is the intent of this report to document the process, in detail, of how the trajectory model is derived. In addition, for accuracies, the relationship and contribution of each partials derivatives and error budgets used in the calculation of Precision Error and MPI error will be explained such that the complex relationships in an indirect ballistic calculation can be determined.

B. BACKGROUND INFORMATION

1. Indirect Fire Trajectory

In **Direct** fire, a target is within the visual range of the shooter and is targeted by the weapon's aiming cue. A direct fire weapon includes line of sight (LOS) weapons, such as infantry small arms and the tank. In contrast, **Indirect** fire engages targets that are out of the shooters' LOS by firing at a high elevation out to long distances. Indirect fire weapons include artillery units, such as the field howitzer, the Multiple Rocket Launcher System (MRLS), and mortars. The role of artillery units is to provide fire support and suppression for the friendly forces. Ammunitions used in an artillery unit are varied and include projectiles, rockets, and missiles. The propellant that provides the propelling force is dependent on the ammunition to be propelled. For instance, solid or liquid propellant in a rocket is contained in the ammunition while the charges used in

mortars and howitzers are located externally. In this thesis, trajectory and accuracies associated with a field howitzer are the focus of interest, thus the projectile would typically be propelled by Bagged Charges or Modular Charge System (MCS).

In an indirect fire attack, historically, most damage is reported to be caused by the initial rounds, and thus the importance of accurate predictions of the trajectories and accuracies. Recent developments on guided munitions, such as the *Excalibur*, improve the accuracy of howitzers that exceeds the best accuracy achieved by indirect fire. The accuracy of an *Excalibur* projectile, unlike conventional dumb munitions, depends solely on the accuracy of its onboard sensors and instruments. Thus traditional factors that affect accuracy such as weather and deviations in launching inputs would not affect the performance of *Excalibur* as compared to dumb munitions. However, the geometric limits of a projectile mean that the high explosive would be limited and the high unit cost of such projectiles would be limited to precision targets in an urban terrain environment.

The trajectory of a projectile fired from a mortar differs from that of a projectile fired from a howitzer. While the trajectory of a mortar is typically characterized by high angle of fire and steep rate of fall, the trajectory of a howitzer is often characterized by lower angle of fire in order to optimize the range and to reduce its time of flight and, consequently, its chance of being detected. In addition, a mortar projectile is generally fin stabilized and a howitzer projectile is spin stabilized. Each stabilization techniques has its their advantages and disadvantages and will be discussed in further sections.

2. Accuracy of Artillery

The prediction of the accuracy of artillery is important in order for the fire support to friendly forces and enemy suppression to be effective. In order to have an accurate prediction, gun factors such as barrel wear and gun condition and external factors such as environment, terrain, and meteorological conditions have to be considered. For instance, when the same projectile is fired from two different barrels with different wear, the nominal trajectories are different; therefore the accuracies associated with the condition of the barrel would be different. This deviation in the barrel condition can be extended to other factors that affect the accuracy of a projectile, and when quantified, can be used to

predict the accuracy given the conditions when the gun is fired. The accuracies are affected by the knowledge of the deviations of the affecting factors.

a. Types of Firing Techniques

In general, the two types of firing techniques in an artillery fire can be grouped under **Unadjusted Fire** and **Adjusted Fire**. **Unadjusted Fire** is also commonly known as **Predicted Fire.**

In **Predicted Fire**, the projectile is fired at a target to achieve a first-round effect without prior firing to correct any cumulative errors. In order to maximize the accuracy of predicted fire, the artillery unit must compensate for any deviations from standard conditions. The errors that must be compensated include the target and launch locations, weapon and ammunition effects and the meteorological conditions. Since accurate targeting can only be carried out based on the availability of information, timely and accurate corrections are essential. In order to compensate for any lack of information, numerous measurements have been undertaken to determine the deviations or error budgets associated with each factor. For instance, the deviations in the muzzle velocity fired from barrels in different conditions can be measured and used in the corrections.

If conditions allow and since standard conditions never exist at any same time, artillery **registration** is conducted to determine the firing data corrections that will be corrected for any cumulative error effects in non-standard conditions. With these corrections, an artillery unit will be able to **transfer** their correction to engage targets within their range in an adjusted fire mode. Errors exist when the transfer is conducted due to the differences in the target location, the time lapsed between the registration and engagement, and ammunition deviations. In modern engagements, registration fire can expose artillery to counter-fire threats and is therefore usually avoided.

b. Types of Errors

The types of errors associated with both adjusted fire and unadjusted fire are the Precision Error (PE) and the Mean Point of Impact (MPI) Error. The PE is the uncorrelated variation about the MPI for a given mission, typically affected by the

ballistic dispersion effect, while the MPI error is associated with the occasion-to-occasion variation about the target, typically affected by the aiming error. The ability to hit the target depends on both PE and MPI errors. In guided munitions, the errors associated are reduced to MPI error since PE would be insignificant due to corrections by the onboard guidance system.

The differences between both errors are shown in Figure 1. The center point is the desired target point. MPI 1 represents the mean impact point of the three shorts in the first occasion. The Precision Error is given by the distance of the shots to the MPI. On the second occasion, due to the variations in the factors affecting the accuracy, the MPI would shift to a new point, MPI 2. Similarly, the MPI would shift to another new point, MPI 3, on the third occasion. However, the PE for both occasions is the same if the ballistics dispersion effect is constant. Thus, the cumulative miss distance is the addition of the PE and the MPI error.

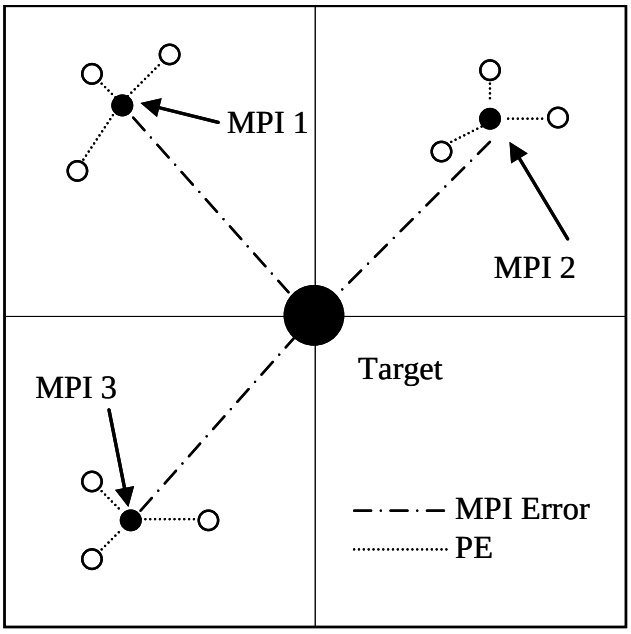

Figure 1. Precision Error and MPI Error

The errors can be resolved into two quantities, the **range** and the **deflection** errors. The range error is the error in the direction between the artillery unit and the target known as the gun-target-line (GTL) while deflection error is in the perpendicular direction to the range direction.

C. STRUCTURE OF THESIS

Chapter II presents the theories of the how the trajectory is influenced by different factors. Chapter II starts with background information on the different phases of the ballistic of a projectile. Since the external ballistics is the most influential factor in the trajectory and the accuracy of an artillery unit, the contributing factors that have an effect on the external ballistics will be discussed in detail.

In Chapter III, the basics of the various theoretical trajectory models and their differences are explained. In particular, the three Degree of Freedom (DOF) Point Mass Model, which is used in the thesis, will be discussed in detail. Other models such as the simplest Zero Drag Model and the computational intensive 5 DOF Modified Point Mass Model will be discussed, including the differences between them.

In Chapter IV, the methodologies for calculating the accuracy are presented. In addition, the importance of the error budgets and unit effects are discussed in detail. Lastly, in this chapter, the methodology for calculating the PE and MPI errors is explained.

Chapter V discusses the trajectory results and how they differ from test data collected from live firing and from the NATO Artillery Ballistic Kernel (NABK). The accuracy results are also compared to results from established software programs, such as Joint Weapon Accuracy Model (JWAM). Discrepancies between the results generated by both methodologies are explained. In addition, the effects on the predicted errors due to how the partial derivatives are calculated will be discussed.

The software implementation of the prediction methodology has been done using Matlab. The description of the software programming is discussed in APPENDIX I.

THIS PAGE INTENTIONALLY LEFT BLANK

II. THEORY OF TRAJECTORY IN ARTILLERY

The trajectory of an indirect fire in artillery can be divided into four different phases: internal ballistics, intermediate ballistics, external ballistics and the terminal ballistics. In each phase, the trajectory can be affected by the ammunition or the environment or, in some cases a combination of both. Since a projectile spends most of its flight in the external ballistics phase, it is subjected to many factors that could alter its original trajectory and affect the accuracy.

A. BALLISTICS PHASES IN ARTILLERY

1. Internal Ballistics

The internal ballistics phase refers to the projectile's behavior from the time the propellant is ignited to the time the projectile exits from the gun barrel. When the propellant is ignited, chemical reactions take place and rapidly generate hot expanding gases. The expanding gases provide the kinetic energy required for the projectile to be propelled.

A projectile fired from a howitzer is spin stabilized, and the spinning effect will be discussed in a later section. The spin rate of a spinning projectile is dependent on the rifling in the barrel. When the projectile is rammed into the barrel chamber, the copper band, or driving band engages onto the rifling of the barrel. As a result, when the projectile is being propelled out of the barrel, the projectile rotates according to the turn rate of the rifling.

Due to the hot gases generated and the friction of the projectile against the bore of the barrel, a barrel will experience a substantial wear rate. The condition of the barrel has an definite impact on the muzzle velocity of the projectile and therefore the accuracy of the artillery unit. However the relationship between the wear rate and both the muzzle velocity and accuracy is not simple. In general, a projectile fired from a barrel with high wear will have lower muzzle velocity due to gas leakages when the propellant is ignited.

2. Intermediate Ballistics

The intermediate ballistics phase refers to the study of the behavior of the projectile in the short transition period between the internal and external ballistics phases. When the projectile exits the barrel, the gases are still expanding. The energies from the expanding gases are dissipated, retained by the gun and transferred to the projectile.

The expanding pressure behind the projectile when it exits the muzzle has a considerable effect on the muzzle velocity of the projectile. The high pressure gases generate a powerful blast shock when the projectile suddenly exits from the barrel and expand rapidly with velocities far greater than the projectile such that a shock wave forms at the base of the projectile. This causes the projectile to gain additional acceleration. In addition to affecting the projectile's muzzle velocity, the expanding gases cause shocks and turbulence due to mixing of the air. The turbulent flow of air can cause adverse yawing of the projectile thus affecting the accuracy.

3. External Ballistics

The external ballistics phase occurs when the projectile has exited the barrel and attained a muzzle velocity and is influenced by the internal and intermediate ballistics. In the external ballistics phase, the main forces acting on the projectile are gravity and air resistance.

The trajectory of the projectile is affected by a number of factors; firstly, the properties of a projectile which include the mass, caliber, geometry and the spin rate, and secondly external environmental effects, such as air density, temperature, and wind speed. Since the external ballistics has the most influence over the trajectory and accuracy, these contributing factors will be discussed in detail in later sections.

4. Terminal Ballistics

Terminal ballistics refers to the study of the interaction of the projectile and the target. The terminal ballistics phase is affected by the impact velocities, the impact angle, the types of projectiles, the fuzes parameters, and the target. The impact velocity of a projectile varies with the barrel elevation. For example given a fixed range of 10

kilometers with a muzzle velocity of 685 meters per second, the projectile can be launched at a gun elevation, or quadrant elevation (QE) of 215 milradian or 1315 milradian. Due to the differences in atmospheric density and the time of flight, the higher quadrant elevation launch is able to gain an increase of about 30 meters per second. However, due to the longer time of flight, the associated accuracy would be degraded.

The impact angle and the angle of attack are not only related to the launch angle but are also closely related to the stability of a projectile. The difference between the impact angle and the angle of attack is shown in Figure 2. Later sections will show that a projectile that is over-stabilized or under-stabilized will have an effect on both angles.

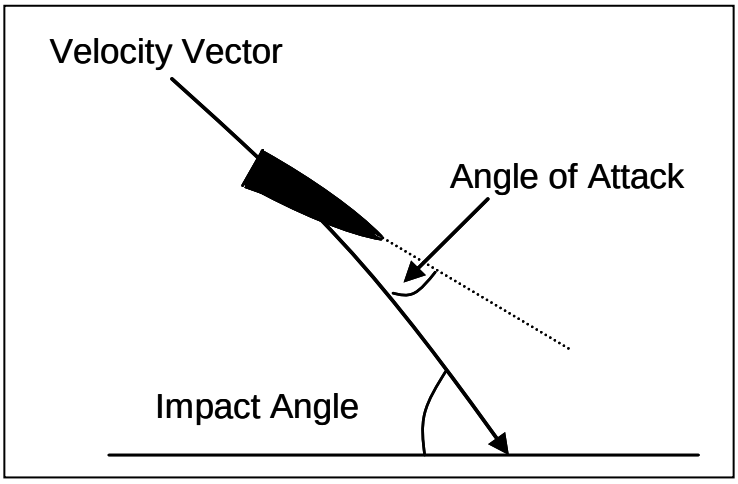

Figure 2. Impact Angle and Angle of Attack

Since the purpose of firing an artillery projectile is varied, from causing infantry casualties to destroying an armored vehicle, the projectiles used are different. The types of projectiles range from high explosive fragmentation to cargo sub-munitions. In order to meet the end effect of the various projectiles, the fuzes used must be programmed accordingly. For instance, when a cargo sub-munitions is used, a proximity fuze must be used so that at a pre-determined time of flight or pre-determined height, the cargo sub-munitions can be dispersed effectively. In addition, there are differences in calculating the accuracy for point detonation or proximity detonation due to factors such as the existence of low-level wind that can affect the movement of the cargo sub-munitions after release.

B. EXTERNAL BALLISTICS

1. Stabilization of Projectiles

Projectiles used in artillery are inherently unstable. A smooth projectile has the CP in a position forward of the center of gravity (CG), hence a sudden disturbance introduced to the projectile will create a disturbance acting about the CG, as shown in Figure 3. Consequently there will be a moment about the CG causing the projectile to be unstable.

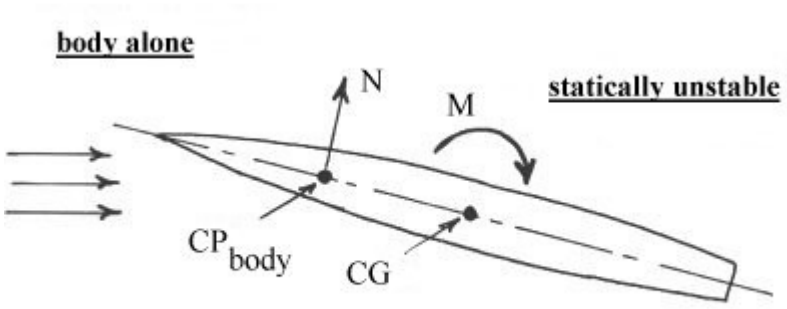

Figure 3. Body Forces on an Unstable Projectile [see ref. 8]

A projectile in flight must be stabilized such that its motion following a disturbance would return to the equilibrium condition before the disturbance. In addition, the trajectory and accuracy of a stabilized projectile can be affected by its reaction rate when returning to the equilibrium position. In general, the stability of a projectile in flight can be achieved from two techniques, spin stabilization and fin stabilization.

a. Types of Stabilization

Fin stabilization, commonly used in mortar projectiles, uses relatively small fins at the rear of the projectiles for stability. The resultant forces from the fins and the body of the projectile act on the center of pressure (CP).

The use of fins at the rear of the projectile increases the surface area behind the center of mass and thus brings the overall CP of the projectile rearward as shown in Figure 4. In this instance, a disturbance can be counteracted by a similar force

generated through the CP to enable the projectile to return to the equilibrium position. The main disadvantage of fin stabilization is that it increases drag on the projectile and is susceptible to large wind effects.

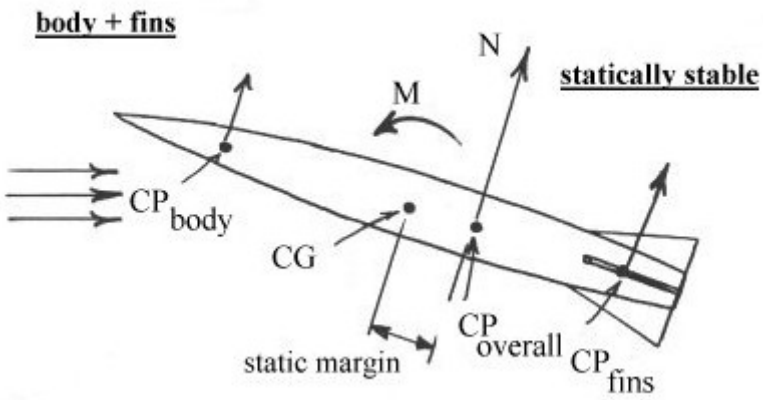

Figure 4. Projectile with Fin Stabilization [see ref. 8]

On the other hand, a **spin stabilized** projectile uses the gyroscopic effect due to the spin rate to achieve its stabilization. The consequence is that perturbing forces are being resisted in the same way as a spinning top due to the angular momentum of the projectile. However, the stability of using spin stabilization is dependent on the spin rate which could cause a projectile to be over or under stabilized.

b. Gyroscopic Effect in Spin Stabilized Projectiles

The reaction of a spinning projectile is similar to when a disturbance is introduced to a spinning top. The CG of the spinning top will be displaced off the vertical axis when a disturbance is introduced. Consequently it generates an overturning moment. The gyroscopic response will be 90 degrees out of phase and will displace the top in a perpendicular plane to the applied moment in which another overturning moment and another 90 degrees out of phase precession will result. This motion continues until the disturbance is damped out of the system through a series of precession.

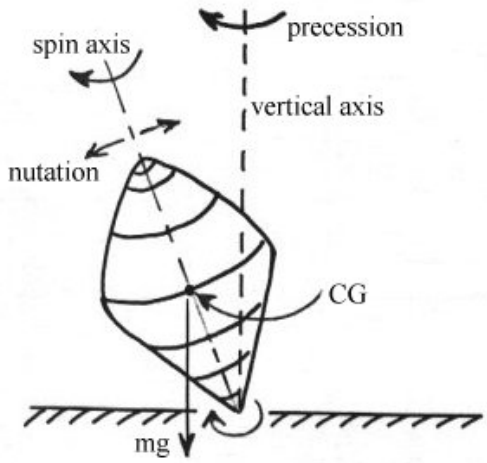

Figure 5. Gyroscopic Stabilization on a Spinning Top [see ref. 8]

The gyroscopic effect on a spinning projectile is the same as the spinning top. When a clockwise spinning projectile is subjected to a vertical disturbance during flight, the resulting motion will cause the projectile to move to the right. Therefore, viewing from the rear of a projectile, the net yaw of the projectile caused the projectile to move in a clockwise direction. This effect is repeated with the consequence of the original disturbance resulting in the nose of the projectile prescribing decreasing amplitude of spiral spin along its flight trajectory as shown in Figure 6. This is commonly known as *nutation*. The resulting flight trajectory would not be in a straight line; rather it would precess along its trajectory, as shown in Figure 7.

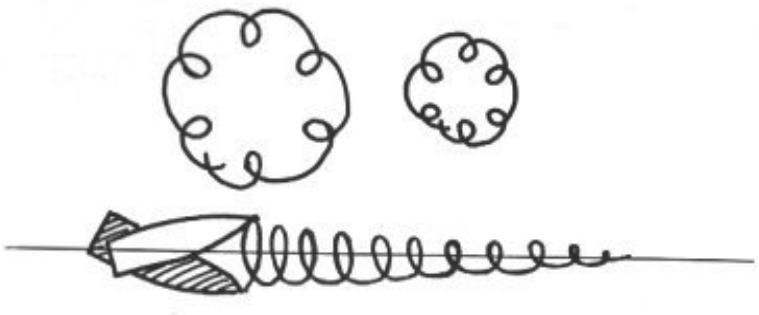

Figure 6. Nutation of a projectile subjected to an initial disturbance [see ref. 8]

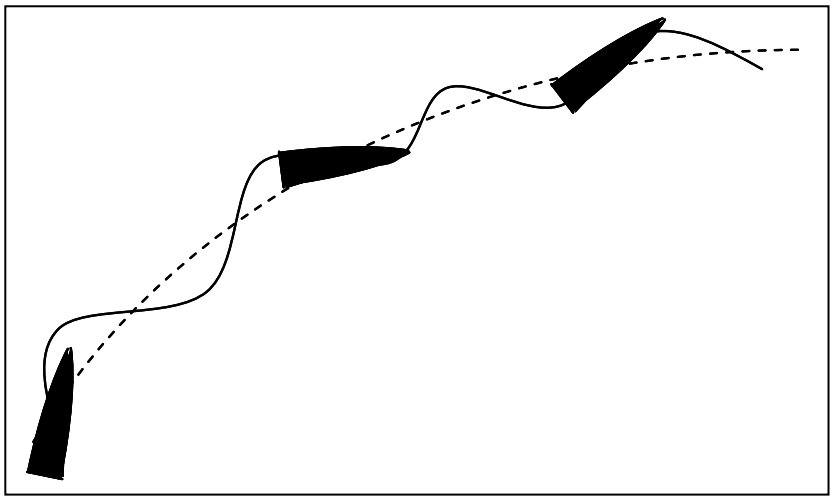

Figure 7. Precession of a Projectile Along its Trajectory

c. Spin Rate Effect

The spin rate of a spinning projectile is determined from the internal ballistics phase characterized by the number of turns per caliber length of the barrel, the barrel length, and the muzzle velocity. In addition, the factors that affect how a projectile precesses along the trajectory depend on the weight distribution of the projectile, its geometry and the location of the CG.

The amount of spin rate for a stabilized projectile is bounded by both the upper and lower bounds of the amount of spin which can be employed. The lower bound refers to the small amount of spin employed which consequently causes the projectile to tumble in flight when there is a disturbance. This is due to a rapid increase in precession causing the projectile to be unstable. On the other hand, when the upper bound is employed, the projectile spins too fast to resist any attempts to perturb it. Thus it precesses so slowly that the nose does not follow the trajectory causing the projectile to land base first. In addition, using spin rates at both the upper and lower bound results in a shorter range. The effects of the spin rate can be seen in Figure 8.

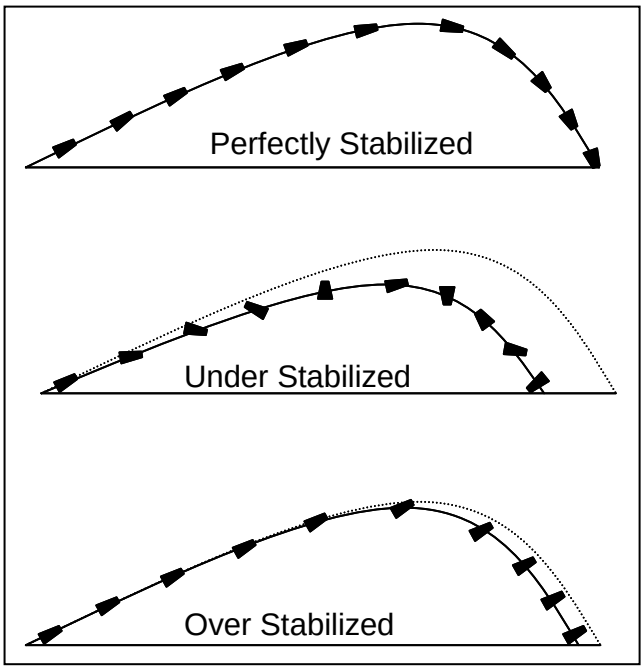

Figure 8. Spin Rate Effect on Stabilization

2. Drag on a Projectile

a. Zero Drag Environment

A zero drag environment represents a projectile's trajectory in a vacuum where the only external force acting on the projectile is gravity which has the effect of pulling the projectile towards center of the Earth. Even though the gravitational constant varies with distance from Earth, the effect on the trajectory of a howitzer shell is small and is assumed to be constant. Thus assuming a flat ground, when the projectile is launched at a muzzle velocity of v_o, the vertical velocity decreases to zero at the apex and increases back to v_o when the projectile impacts the target. In the horizontal plane, since there is no resistance to the motion other than gravity, the horizontal velocity component is constant.

Distinct features of a zero drag trajectory include the following:

i. Equal launch and impact velocities.

ii. Maximum range at exactly forty-five degrees.

iii. Equal launch and impact angles.

iv. The trajectory is symmetrical about the vertical line through the apex.

v. The trajectory remains in the vertical plane that contains the line of departure.

b. Aerodynamic Drag

In a real fluid such as the Earth's atmosphere, an additional force caused by the resistance of the fluid on the projectile will introduce an opposing force to the projectile; this is known as the *drag*. The three contributors to the aerodynamic drag are the **skin friction**, **base drag** and the **wave drag**. The skin friction drag is caused by the resistance of the fluid and the surface of the projectile. The base drag is caused by the air turbulence causing a pocket of low air pressure behind the projectile and is a function of base area. Lastly, the wave drag is caused by the compression and expansion of air as it travels over the projectile and is dependent on the shape. The three drag components can be added together to form the total aerodynamic drag coefficient,

$$C_D = C_{D(\text{wave})} + C_{D(\text{friction})} \frac{A_{\text{wet}}}{S_{\text{ref}}} + C_{D(\text{Base})} \frac{A_{\text{base}}}{S_{\text{ref}}}, \tag{2.1}$$

where $C_{D(\text{wave})}$: Wave drag coefficient,

$C_{D(\text{friction})}$: Friction drag coefficient,

$C_{D(\text{Base})}$: Base drag coefficient,

A_{wet}: Wetted area of the projectile,

S_{ref}: Reference area of the projectile, and

A_{base}: Cross-sectional area of the projectile.

With the total drag coefficient known, the drag force acting on the projectile can be written as

$$F_D = \frac{1}{2} S_{\text{ref}} \rho v^2 C_D, \tag{2.2}$$

where F_D: Drag force,

ρ: Density of air, and

v: Velocity of projectile.

The drag coefficient is a measure of the efficiency of the projectile to reduce its resistance to the fluid and since it is a function of both the Mach number and the shape of the projectile. For a given projectile, the drag characteristics of a projectile can be represented by a curve of drag coefficient against the Mach number. The drag coefficient reaches a peak in the region of Mach 1 where it is trying to break through the speed of sound and reduces to almost constant in the supersonic region. The drag coefficient used in this thesis and the drag curve is as shown in Appendix B.

c. Effects of Drag

The drag force is a function of the absolute relative speed of a projectile and can substantially modify the trajectory of a projectile as compared to a vacuum. The launch and impact velocities differ from one another, as do the launch and impact angle.

The most profound effect of the drag on a projectile is the reduction in range. For a given muzzle velocity and quadrant elevation, the range calculated using a zero drag environment can be almost twice of the range when drag is present, as shown in Figure 9. In addition, in the presence of drag, the trajectory is longer symmetric, as in the zero drag environment case. This is due to the fact that as the projectile descends from the apex, the drag force on the projectile increases causing the projectile to fall at a steeper angle. Consequently the projectile impacts the earth at an impact angle greater than the launch angle.

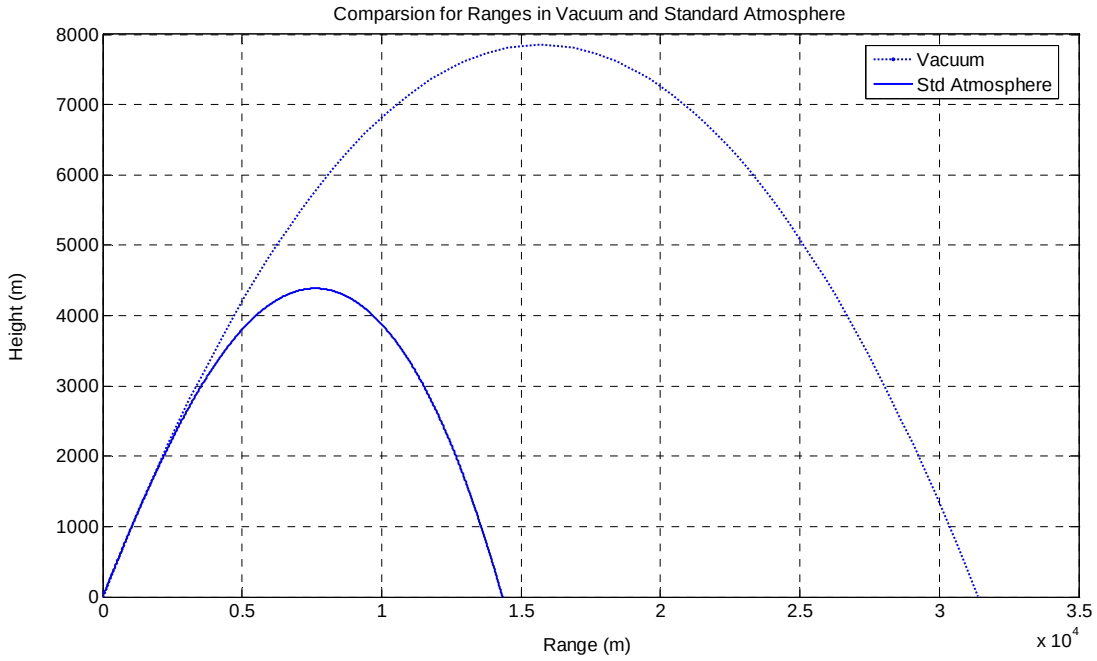

Figure 9. Range comparison between zero drag and real fluid

In order to reduce the adverse effect of drag, attempts have been made to reduce the drag force acting on the projectile. Examples are introducing base bleed and boat tail designs such that the base drag can be reduced. In addition, reducing the wetted area of a projectile would reduce the friction drag.

3. Trajectory Due to Spinning Projectile

In a standard atmosphere where there is no wind, the trajectory of a projectile fired from a gun is a parabola shape when projected in a vertical plane. This is caused by the gravitational force acting on the projectile after it leaves the gun. However, when the trajectory is projected on a horizontal plane, the trajectory is also a curve, due to gyroscopic properties caused by the spinning projectile. When the projectile is spinning clockwise viewed from the rear, the drift will be to the right and vice versa.

a. Gyroscopic Effect on Drift

The stability of a spinning projectile is achieved by the gyroscopic effect. However, the gyroscopic effect also causes the projectile to drift in deflection. In an

earlier section, the nutation of a spinning projectile was discussed and the net result is that the nose precesses around the trajectory, as shown previously in Figure 7. However, at the same time, the trajectory is also dipping, and through a combination of spin rate, precession, and a dipping trajectory, the yaw of the projectile is almost constant. Thus the projectile moves along the trajectory with its center of gravity on the trajectory but the nose rosettes about the trajectory with an average position off to the right. This average yaw is also known as the yaw of repose or equilibrium yaw.

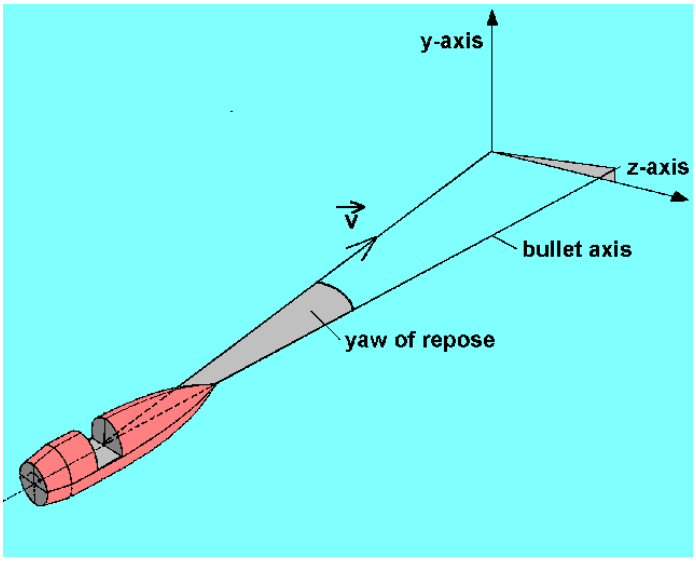

Figure 10. Yaw of repose in a projectile's trajectory [see ref. 9]

With the projectile yawing to the right, the air stream will create a higher pressure on the exposed left side. Similar to an airfoil, the pressure difference on both sides of the projectile will attempt to push the projectile to the right. A clockwise spinning projectile will always experience a drift to the right. On the other hand, an anti-clockwise spinning projectile will have a yaw of repose to the left of the trajectory and thus cause the projectile to always drift to the left.

b. Magnus Effect on Drift

The Magnus effect is the physical phenomenon where the rotation of a projectile affects its trajectory when traveling through a fluid. The higher velocity above

a rotating body indicated by the closer streamlines is reflected by a reduction in pressure. On the other hand, the lower velocity underneath the rotating body has a higher pressure. The net effect of these pressure changes produces a lift on the body and an increase in range.

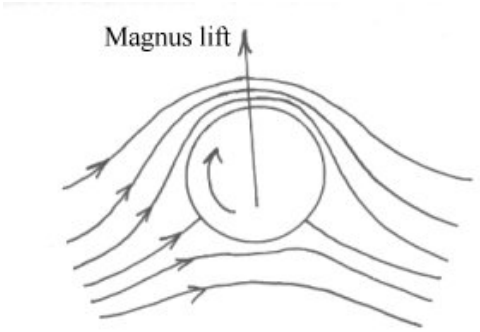

Figure 11. Magnus Effect on a Rotating Body [see ref. 8]

Since the projectile is traveling with the nose displaced to the right, also known as equilibrium yaw, there is a cross flow of air from left to right of the body. Thus the top and left sides of the projectile will have a higher velocity than the bottom and right sides of the projectile. Consequently, when the projectile is flying at an equilibrium yaw, it will experience an average Magnus lift due to the pressure difference between the top and bottom of the projectile which increases the range. In addition, since there is a cross flow of wind from left to right causing pressure differences as indicated by the closer streamlines on the left in Figure 12, the Magnus effect would tends to pull the projectile to the left, opposing the gyroscopic forces.

Golf balls are dimpled such that when a back spin is applied, the dimples on the bottom of the golf ball retain pockets of turbulence, thus causing the pressure to increase. This increases the lift of the golf ball and the range considerably. However, due to the smooth surface of the projectile, the lift generated by the Magnus effect is small. Similarly, the drift caused by the Magnus effect is small, due to a small angle of repose as compared with the gyroscopic effect and thus the Magnus effect in range and deflection calculation is often ignored.

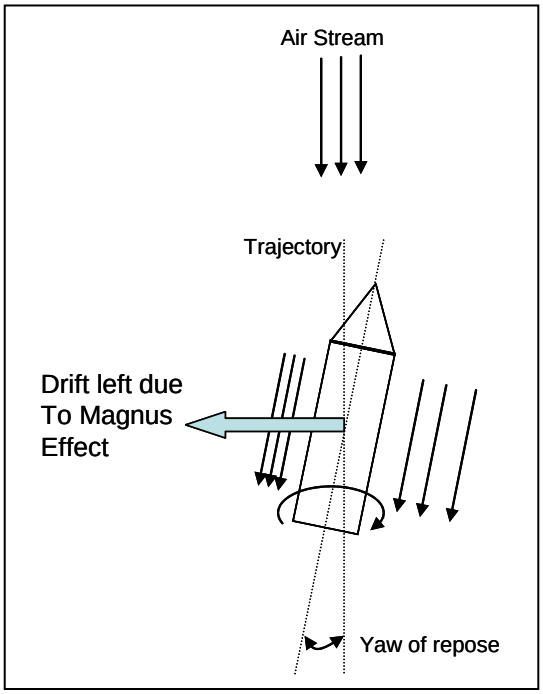

Figure 12. Magnus Effect of a Rotating Body Looking from the Top

4. Coriolis Effect

The Coriolis Effect is the deflection of a moving object in a rotating frame of reference and is caused Earth's rotation. Since the motion of the Earth's surface is not in a straight line, the target attached to the Earth's surface would have drifted away according to the Earth's rotation while the projectile is still in flight.

The drift due to the Earth's rotation is magnified by both time of flight and target range. Long range missiles, such as ballistics missiles, are severely affected by this drift. The drift of an artillery projectile is much less, with drift of about 100 meters for a range of 20 kilometers depending on the geographical locations. When the range is further reduced to about 5 kilometers, the drift is often less than the PE of a projectile and thus can be ignored without any corrections.

The amount of drift is dependent of the geographical location. For instance, when the projectile is fired vertically in the North Pole, the projectile will fall directly back into the barrel since the Coriolis effect is zero. However, when the same projectile is fired when at the equator, the projectile will land to the east of the barrel. Similarly, when

firing east, in the same direction of Earth's rotation, a projectile will hit a point beyond the target since the earth effectively rotates down as shown in Figure 13.

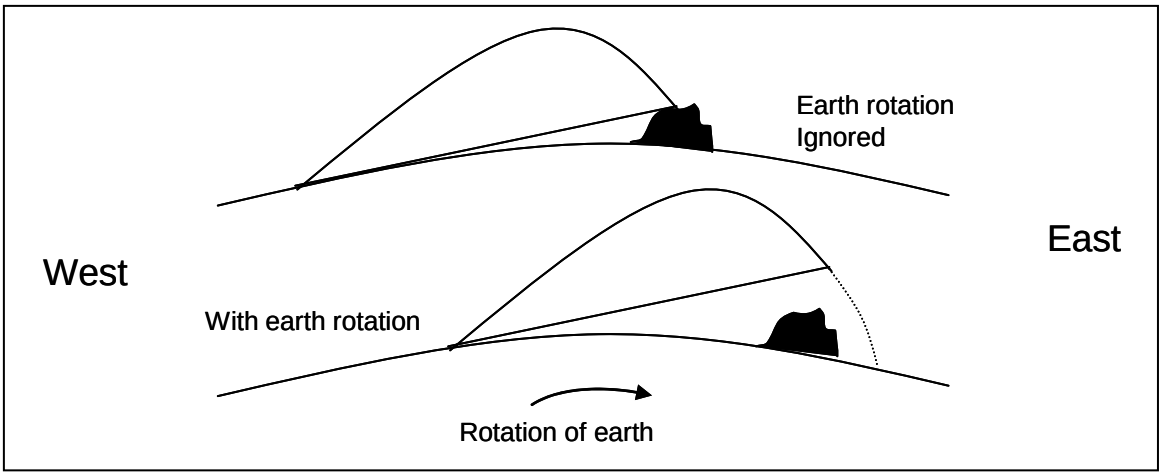

Figure 13. Range Difference when Firing East [see ref. 3]

Since the Earth's dimensions and rotation rate are known, it is possible to estimate a correction on the trajectory based on the geographical locations of the launching point and the target. The analytical treatment of the drift due to Coriolis Effect is complex and implementation of the effect in computational software is rigorous. In addition, once the correction is applied to the firing solution, the Coriolis Effect does not have an effect on the accuracies. Thus it is assumed in this thesis that corrections have already been done and there is minimal effect of the Coriolis effect on the accuracy.

C. ATMOSPHERIC EFFECTS

1. Wind Effects

The effect of the wind, **range wind** and **cross wind**, on both the trajectory and accuracy of a projectile is significant. It is rare to see cases where either the range wind only or the crosswind only affects the trajectory; it is most often a combination of the two winds. However, when studying wind effects, it is easier to study how range and crosswind affect the trajectory individually. In the artillery units, the wind components are always resolved into the range and crosswind components and corrections are made

to counter the two components. The effect of the crosswind is not trivial, a crosswind does not only contribute only to the drift, it also has an effect on the range of a projectile.

The easiest case in the study of the wind effects is where there is range wind only. Intuitively, a tail wind will increase the range and a head wind will reduce the range. The relative velocity of the projectile increases when there is a head wind and reduces when there is tail wind. When the relative velocity is reduced, the drag force will be reduced according to the aerodynamic drag equation.

When a projectile is flying into a crosswind environment, the projectile will turn into the direction of the wind. The crosswind component will introduce drag onto the projectile by increasing the relative velocity of the projectile. If the net thrust is negligible, that is when the thrust is equal to the drag, the projectile will land on the target, as shown in Figure 14. On the other hand, if an artillery rocket with propulsion is used so that it effectively has a larger thrust than drag (T > D), it would pass upwind of the target. Lastly, in the case of projectiles that have no thrust such as projectiles fired from howitzers, the drag will be greater than the thrust (D > T) and the deviation occurs in the direction of the crosswind

The effect of the crosswind on drift is not as simple as multiplying the crosswind speed with the time of flight If a projectile launched into a cross-wind of 5 meters per second and flew for 100 seconds, it would experience a down wind drift of 500 meters which would be erroneous. In order to calculate the drift caused by the crosswind, the angle in which the projectile turns into the crosswind must be considered. From Figure 14, the relative wind component, which is the vector sum of both the velocity of the projectile and the crosswind component, must be used to calculate the drag force. This drag can then be resolved into two components, the range and the deflection components. Assuming there are no other significant deflection forces acting on the projectile, the deflection component of the drag can be used to determine the amount of drift caused by the crosswind.

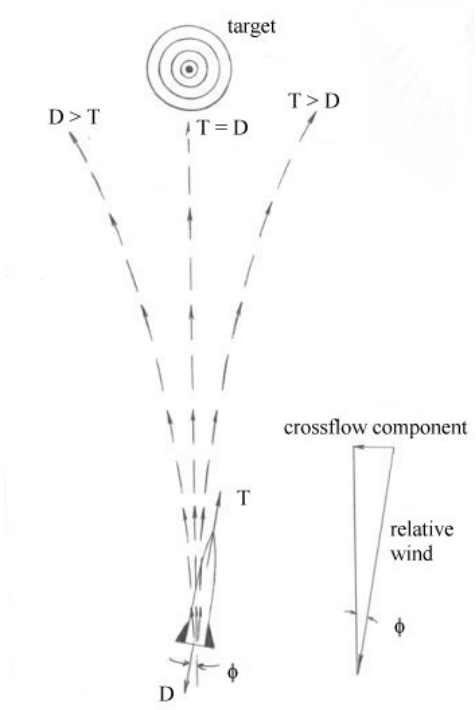

Figure 14. Effects of Crosswind on Drift [see ref. 8]

2. Meteorological Effects

The standard atmospheric data used in this thesis are adapted from the International Civil Aviation Organization. It approximates the average atmospheric conditions in Continental Europe and North America. Under standard atmospheric conditions, accurate fire can be placed onto the target without any adjustment with regards to the meteorological conditions. However, standard atmospheric conditions never exist. The set of standard conditions in artillery is as shown in Table 1.

WEATHER	**STANDARD CONDITIONS**
1	AIR TEMPERATURE 100 PERCENT (59°F)
2	AIR DENSITY 100 PERCENT (1,225 gm/m^3)
3	NO WIND
POSITION	**STANDARD CONDITIONS**
1	GUN, TARGET, AND MDP AT SAME ALTITUDE
2	ACCURATE RANGE
3	NO ROTATION OF EARTH
MATERIAL	**STANDARD CONDITIONS**
1	STANDARD WEAPON, PROJECTILE AND FUZE

2	PROPELLANT TEMPERATURE (70°F)
3	LEVEL TRUNNIONS AND PRECISION SETTINGS
4	FIRING TABLE MUZZLE VELOCITY
5	NO DRIFT

Table 1. Standard Conditions in Artillery [see ref. 11]

Variations in meteorological conditions have an effect on the projectile traveling through the atmosphere and hence affect its trajectory. The artillery projectile typically has peak altitudes of about 20 kilometers which is within the troposphere and is thus subjected to air density and drag. With increasing altitude, air properties such as density, temperature, pressure and air viscosity change. In addition, the air properties are differ by geographic location. The variation in density and temperature with height in standard meteorological conditions is as shown in Figure 15 and Figure 16.

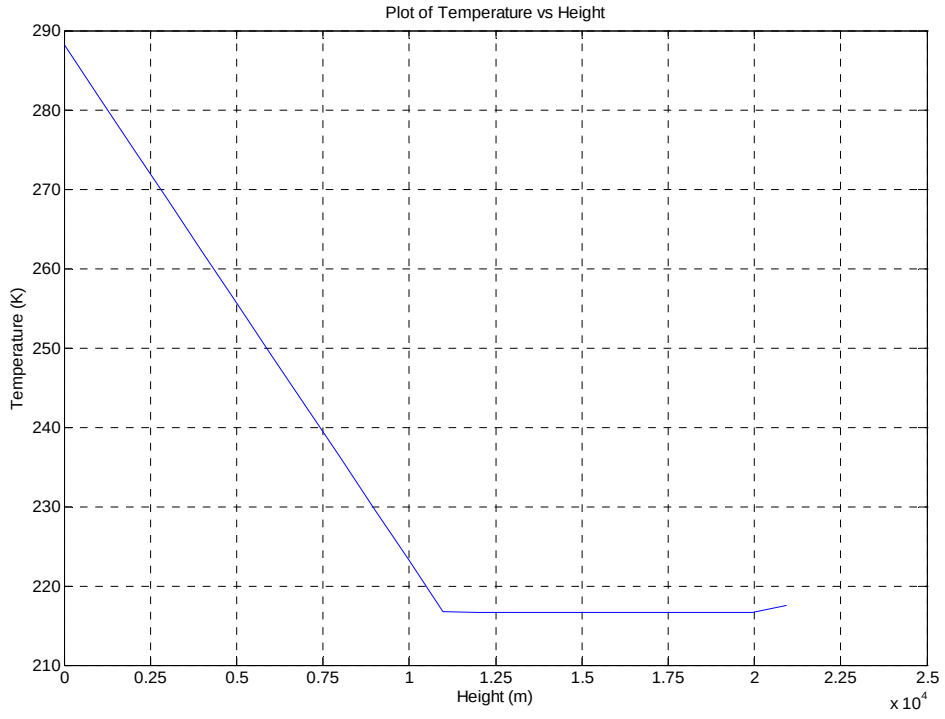

Figure 15. Variation of Temperature with Height

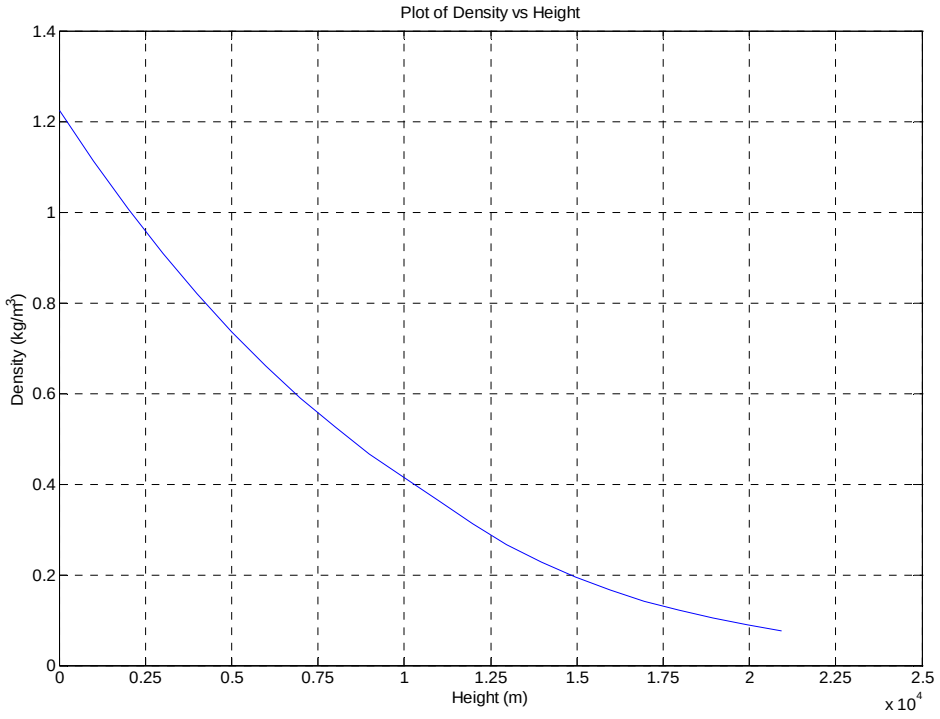

Figure 16. Variation of Density with Height

The acceleration of the projectile is dependent on the drag force of a projectile:

$$F_D = \frac{1}{2} S_{\text{ref}} \rho v^2 C_D \tag{2.3}$$

$$\rho = f\left(\text{Height}\right)$$

$$C_d = f\left(\text{Mach}\right) = f\left(v_s\right) = f\left(\gamma, R, T\right)$$

$$T = f\left(\text{Height}\right)$$

It can be seen that the density is a function of height, and the drag coefficient is influenced by the speed of sound, which is dependent on the temperature, specific gas constant and the adiabatic index. As the speed of sound varies with temperature, the Mach number will vary not only with the absolute velocity of the projectile but with increasing height too. Consequently, a projectile at a greater height would experiences

less drag for the same absolute velocity. In addition, the wind speed and direction will affect the relative velocity of the projectile as explained in an earlier section, and thus affects the drag force.

In an artillery unit, the meteorological conditions are supplied to the fire control system in the form of a met message. The met message contains three properties that would affect the trajectory, the wind speed and direction, the air temperature, and the air density. The met message contains 16 lines with each line representing the weighted average of the atmospheric conditions in that zone up to the height indicated. Table 2 shows a sample of the ballistic meteorological message.

Zone Height	Line Number	Zone Values			
		Wind Direction (10's mils)	Wind Speed (Knots)	Temperature (% of Std)	Pressure (% of Std)
Surface	00	302	04	042	910
200	01	210	12	050	902
500	02	255	10	019	904
1000	03	460	30	018	950
1500	04	421	20	020	930

Table 2. Sample of Ballistic Meteorological Message

The first column in Table 2 indicates the height of the zone where the meteorological conditions are valid, while the second column identifies the altitude zone. The third column and fourth columns indicate the wind direction and speed. The fifth and last columns indicate the variation percentage from the standard atmospheric condition.

The example in line number 03, the zone which covers a height from the surface to 1,000 meters; the wind direction is read as 460 x 10 = 4600 milradians in the clockwise direction from the true north; wind speed is 30 knots; the temperature is 100% + 18% = 118% of the standard condition; and the density is 95.0% of the standard condition.

The standard atmospheric conditions are implemented into the software. In the implementation of the meteorological conditions, the wind speed and direction, and the variations in the density and temperature from the standard conditions are taken into consideration in the prediction of the nominal trajectory. The selection of the line number

from the met message is dependent on the height of the apex of the projectile. For instance, if the apex of a projectile is at 1,500 meters, the line number selected would be 04 from the met message.

THIS PAGE INTENTIONALLY LEFT BLANK

III. TRAJECTORY MODELS

The trajectory of a projectile can be modeled using different methodologies. The common methodologies are the **Zero Drag Model**, the **Point Mass Model,** and the **Modified Point Mass Model**.

The zero drag point mass model is the simplest trajectory model since it calculates the trajectory based on the kinematics of a point mass. Since the drag force is ignored, the main disadvantage associated with a zero drag model is that the trajectory prediction result of a howitzer-fired projectile is poor. Consequently, the zero drag model would predict a range that is greater than what is realistic. On the other hand, the zero drag model is reasonably accurate for calculating trajectories for ballistic missiles, which predominantly spend most of their flight time outside Earth's atmosphere where the only force acting on the missile is the Earth's gravity, and for low drag, slow speed munitions such as free-fall bombs.

The point mass model, which is used in this thesis, takes into consideration the drag and environmental effects and is able to provide relatively accurate results with limited computing capacity. The trajectory prediction can be further improved with increasing degree of freedom (DOF) in the point mass model. The simplest point mass model is the two degree of freedom (2 DOF) model which has the drag and the gravity components. The 2 DOF can be enhanced by the inclusion of the deflection motion. On the other hand, the modified point mass model is complex. It has five degree of freedom but is capable of predicting the trajectory with good accuracies. However a modified point mass model requires more computing resources.

A. ZERO DRAG MODEL

The zero drag point mass model is the simplest trajectory model. It describes the trajectory path of an artillery projectile, since the only force acting on the projectile is the gravity, as shown in Figure 17.

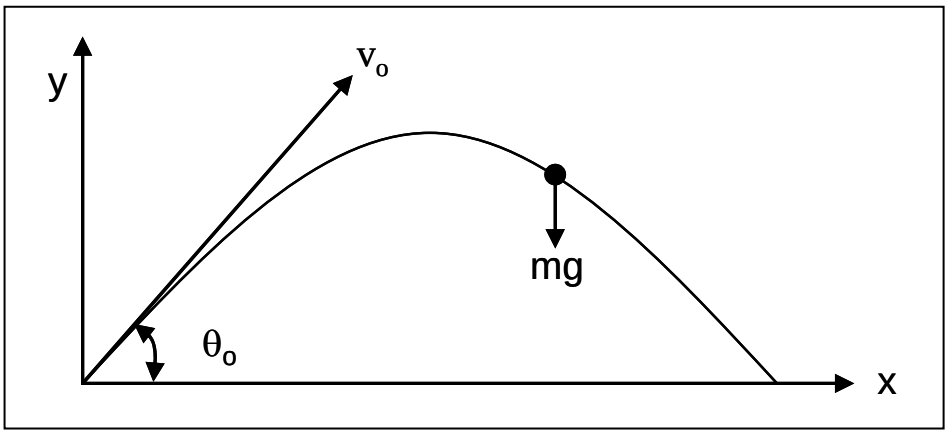

Figure 17. Trajectory of a Projectile in a Zero Drag Model

The zero drag model uses basic kinematics formulae to calculate the velocity, range and time of the trajectory. The initial conditions are given as the following:

$$t = 0,\ x = 0,\ y = 0,\ v_{ox} = v_o \cos\theta_o,\ v_{oy} = v_o \sin\theta_o,$$

where t: Time,

x: Horizontal distance,

y: Height,

v_o: Muzzle velocity, and

θ_o: Quadrant elevation.

Since the only external force is gravity, at any time, t, the horizontal and vertical displacements are given by,

$$x = v_o \cos\theta_o t, \tag{3.1}$$

$$y = v_o \sin\theta_o t - \frac{gt^2}{2}. \tag{3.2}$$

Finally, the range and time of flight at the impact point can be derived as:

$$t_f = \frac{2v_{oy}}{g}, \tag{3.3}$$

$$x_f = \frac{2v_{oy}v_{ox}}{g} = \frac{v_o \sin 2\theta}{g}. \tag{3.4}$$

B. POINT MASS MODEL

1. Two Degree of Freedom Point Mass Model

In the 2 DOF model, it is assumed that the projectile's axis is aligned to the trajectory and the only forces acting on the projectile are the weight and the zero yaw drag. Since the meteorological conditions affect the drag force and thus the acceleration of the projectile, when the meteorological condition is included, the range prediction improves. Since it is assumed that the projectile is perfectly aligned to the trajectory, the crosswind will have no effect on the range calculation of the projectile.

The initial conditions of the point mass model are the same as the zero drag model. However, instead of using kinematics for the trajectory, the body forces are used to derive the acceleration equation. The forces acting on a projectile in a 2 DOF model are shown in Figure 18.

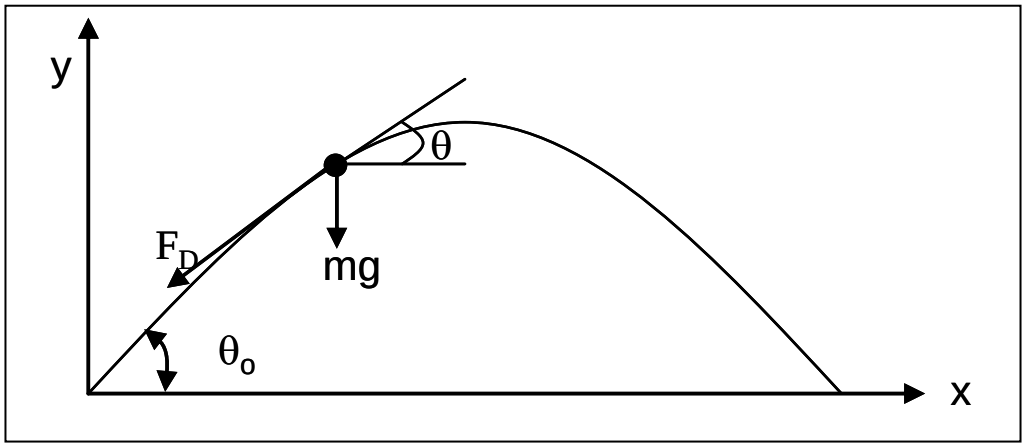

Figure 18. Forces Acting on the Projectile in a Point Mass Model

Since there is zero thrust on the projectile, the drag force is the only axial force experienced by the projectile. Therefore, the horizontal and vertical acceleration of the projectile at time *t* is given by

$$a_x = \frac{-F_D \cos\theta}{m}, \tag{3.5}$$

$$a_y = -g - \frac{F_D \sin\theta}{m}, \tag{3.6}$$

where a: Acceleration of the projectile,

θ: Angle between the velocity vector and horizontal plane,

g: Gravitational constant, and

m: Mass of projectile.

The velocity of the projectile can be evaluated by integrating the acceleration:

$$a = \frac{dv}{dt} \approx \frac{v(t+dt) - v(t)}{dt}, \tag{3.7}$$

where v: Velocity of the projectile.

Therefore, for a given time step, *dt*, the horizontal and vertical component of the velocity can be evaluated by

$$v_x(t+dt) = a_x dt + v_x, \tag{3.8}$$

$$v_y(t+dt) = a_y dt + v_y. \tag{3.9}$$

Similarly from the velocity equations, the horizontal and vertical component of the displacement can be derived as

$$X(t+dt) = v_x dt + X, \tag{3.10}$$

$$Y(t+dt) = v_y dt + Y \tag{3.11}$$

where X: Displacement in the range direction, and

Y: Displacement in the vertical direction.

The accuracy of the trajectory prediction is dependent on the time step, and the prediction accuracy improves when smaller time steps are used.

2. Modified Point Mass Model

The modified point mass model is a compromise between a simple point mass model and a computationally intensive 6 DOF point mass model. In the modified point mass model, the effects due to the spin rate of a projectile are included. Thus the equilibrium yaw angle in both the lateral and trajectory plane is taken into account for calculation of the drift and drag. The modified point mass model is implemented in trajectory programs such as the NATO Armaments Ballistics Kernel (NABK) and the Battlefield Artillery Target Engagement System (BATES).

Even though it was mentioned in earlier sections that the Magnus effect is very small, the Magnus moment following a disturbance generates an incremental nose up moment which is opposed to the gyroscopic effect. This subsequently lead to instability of the projectile. The Magnus effect on a projectile increases with quadrant elevation and leads to an increase in the yaw angle. Thus the Magnus effect could cause the projectile to drift to the left, countering the gyroscopic effect.

In the 2 DOF model, the acceleration equation contains only the drag and gravity terms. However, in the modified point mass model the lift, the Magnus force, and the Coriolis acceleration are included in the acceleration equations. In addition, the trajectory in the range and deflection is coupled, making the computation complicated. The acceleration of the projectile using the modified point mass model is given by the vector equation

$$\vec{\ddot{u}} = \vec{D} + \vec{L} + \vec{M}_m + \vec{g} + \vec{\Lambda}\,, \tag{3.12}$$

where D: Drag force vector,

L: Lift force vector,

M_m: Magnus moment, and

Λ: Coriolis acceleration.

The evaluation of each term is not trivial and the resulting equations can be found in NATO STANAG 4355 Modified Point Mass Trajectory Model.

3. Three Degree of Freedom Point Mass Model

The 3 DOF model is an improvement over the 2 DOF model since instead of predicting the trajectory in the vertical plane only, the 2 DOF model is modified to consider the trajectory of a projectile in the horizontal plane. Instead of using the modified point mass model in which the deflection and range trajectories are coupled, the 3 DOF model is decoupled and treat the range and deflection trajectories separately. Thus the wind effects not only affect the range as in the 2 DOF model, but the deflection direction as well.

Inevitably, since the Magnus effect and the Coriolis effect are ignored, the trajectory prediction is less accurate as the quadrant elevation increases. This difference in the trajectory results between a 3 DOF model and a modified point mass model will be discussed in Chapter V.

The acceleration, velocity and displacement equations are the same when compared with the 2 DOF model. The differences are due to the inclusion of the relative velocity of the projectile and the velocity of the wind. The relative velocity of the projectile along its trajectory is not a scalar summation of the projectile velocity and the range wind, but a vector summation, as shown in Figure 19. Thus, the relative velocity of the projectile is

$$\vec{v}_r = \vec{v} + \vec{w}, \tag{3.13}$$

where $\vec{v}_r$: Relative velocity of the projectile, and

$\vec{w}$: Velocity of the wind.

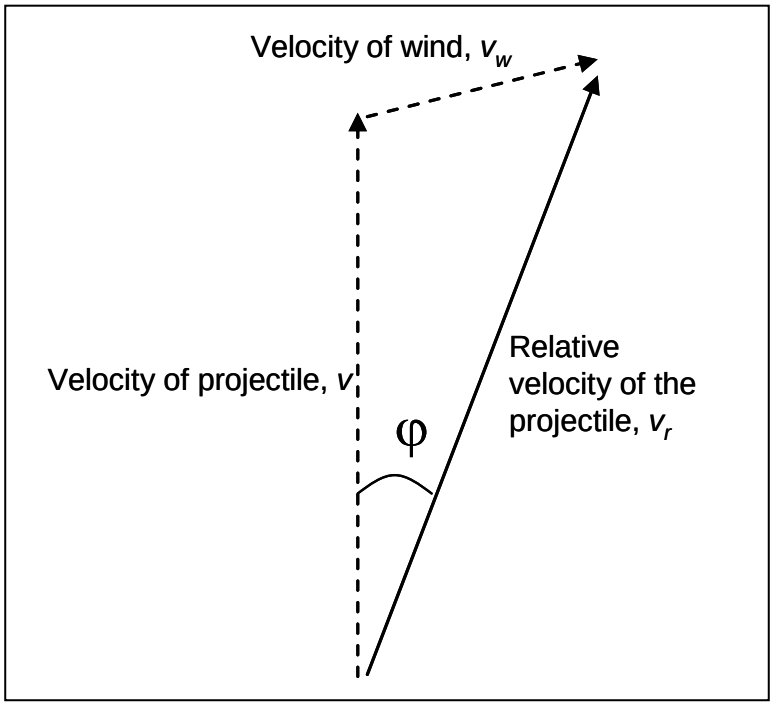

Figure 19. Relative Velocity of the Projectile

a. Drift due to the Wind

The total drag force with the addition of the crosswind acting on the projectile is

$$F_D = \frac{1}{2} S_{ref} \rho v_r^{\,2} C_D \,. \tag{3.14}$$

Consequently, the drag force can be resolved into two components, the range and deflection components:

$$F_{D,R} = F_D \cos\varphi \,, \tag{3.15}$$

$$F_{D,D} = F_D \sin\varphi \,. \tag{3.16}$$

where $F_{D,R}$: Drag force in the range direction,

$F_{D,D}$: Drag force in the deflection direction, and

φ: Angle between the projectile axis and trajectory.

The range component of the drag force, $F_{D,R}$ is used to calculate the acceleration components in the range direction and the deflection component is used to calculate the acceleration due to the crosswind. Thus the horizontal and vertical components of the acceleration on the projectile in range can be derived as

$$a_x = \frac{-F_{D,R}\cos\theta}{m}, \tag{3.17}$$

$$a_y = -g - \frac{-F_{D,R}\sin\theta}{m}. \tag{3.18}$$

The acceleration of the projectile in the deflection direction is

$$a_{D,w} = \frac{F_{D,D}}{m}. \tag{3.19}$$

Therefore, the drift due to the wind effect can be written as

$$Z_w = \frac{1}{2}a_{D,w}t^2. \tag{3.20}$$

b. Drift due to Rotating Projectile Effects

It was explained in earlier sections that the projectile will experience drift due to the spin rate. There are several ways to calculate the drift of the projectile; the most accurate method uses the modified point mass model. However, empirical formulas are available and have been traditionally used in the calculation of drift of projectiles used in the Navy gun. In the NAVORD Report No. 5136, the drift before World War II was computed according to

$$Z_p = K^* t^2, \tag{3.21}$$

$$K^* = \frac{(3.5)^2}{\left[0.871 + \frac{349.4n}{v_o} + \frac{\theta}{128.2}\right]L^2}, \tag{3.22}$$

where Z_p: Drift due to rotating projectile,

θ: Angle of elevation,

v_o: Initial velocity in feet per second,

l: Length of the projectile in calibers,

n: Length of gun per turn of rifling, and

t: Time of flight in seconds.

However, the units for the angle of elevation and the length of gun per turn of rifling are unknown. Using this formula, the drift cannot be determined with certainty in this thesis.

Instead of using the modified point mass model and the empirical formula in NAVORD Report No. 5136 to calculate the drift, a simple method is used in this thesis to estimate the drift. Since the projectile is spinning, it has an inherent lateral acceleration in the deflection direction. However, since the spin rates decrease with time, the acceleration is not constant and thus the trajectory on the horizontal plane is a parabola, as shown in Figure 20.

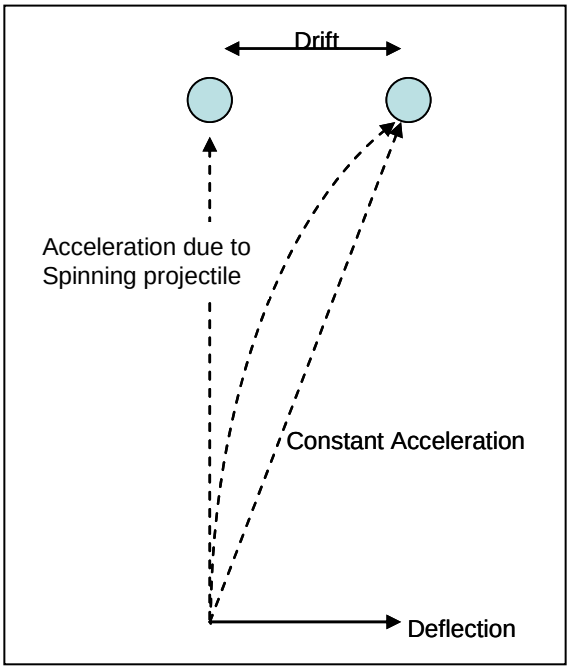

Figure 20. Drift Caused by Spinning Projectile

If a constant acceleration can be assumed, as shown in Figure 16, the drift due to the projectile effects can be estimated by

$$Z_p = \frac{1}{2} a_{D,p} t^2, \tag{3.23}$$

where $a_{D,p}$ is the estimated cumulative lateral acceleration of the projectile due to the gyroscopic effect determined from actual data for a particular projectile. A sample of how the acceleration is calculated is shown in Appendix C.

The total deflection experienced by a projectile using the 3 DOF model is,

$$Z = Z_w + Z_p \tag{3.24}$$

IV. ERROR CALCULATIONS

It was mentioned in Chapter I that the PE and the MPI errors are different for adjusted fire and predicted fire, and the objective of this thesis is to formulate the methodology for the calculation of the PE and the MPI errors for predicted fire. In addition, it was mentioned in Chapter II that standard conditions do not exist, thus there will always be variations in firing conditions. Consequently, the different firing conditions lead to different errors. In order to verify the accuracy of the results from the methodology used in this thesis, the error budgets in this thesis are referenced to the error budgets used in JWAM, such that the accuracy results can be compared in Chapter V.

A UNIT EFFECTS/PARTIAL DERIVATIVES

The unit effects are calculated by using the 3 DOF trajectory model. A simple error margin in accuracy can be illustrated, for example, due to the variation in the muzzle velocity. Since the muzzle velocity of a projectile depends on a number of factors, such as the mass of a projectile, the variation in the propellants, and the barrel condition, there will always exist a variance in the muzzle velocity, σ_v. At the impact point, the variation in the range due to the variation in the muzzle velocity can be written as

$$\sigma_{X,v} = \left(\frac{\partial X}{\partial v} \right) \sigma_v , \tag{4.1}$$

where $\frac{dx}{dv}$ is the unit effect due to muzzle velocity and is related to the difference in range caused by a difference in the muzzle velocity. Thus the unit effect can be approximated as

$$\frac{\partial X}{\partial v} = \frac{X_2 - X_1}{v_2 - v_1}, \tag{4.2}$$

where the ranges X_2 and X_1 are related to the muzzle velocities v_2 and v_1 respectively. The unit effects are calculated independently by keeping all other variables and inputs constant. Figure 21 shows an example of the QE plotted with a constant muzzle velocity

of 684.5 meters per second. It shows that the unit effect, which is the slope of the graph at a particular point, is not constant and it varies with range. Therefore, at every launch angle, the unit effect is different.

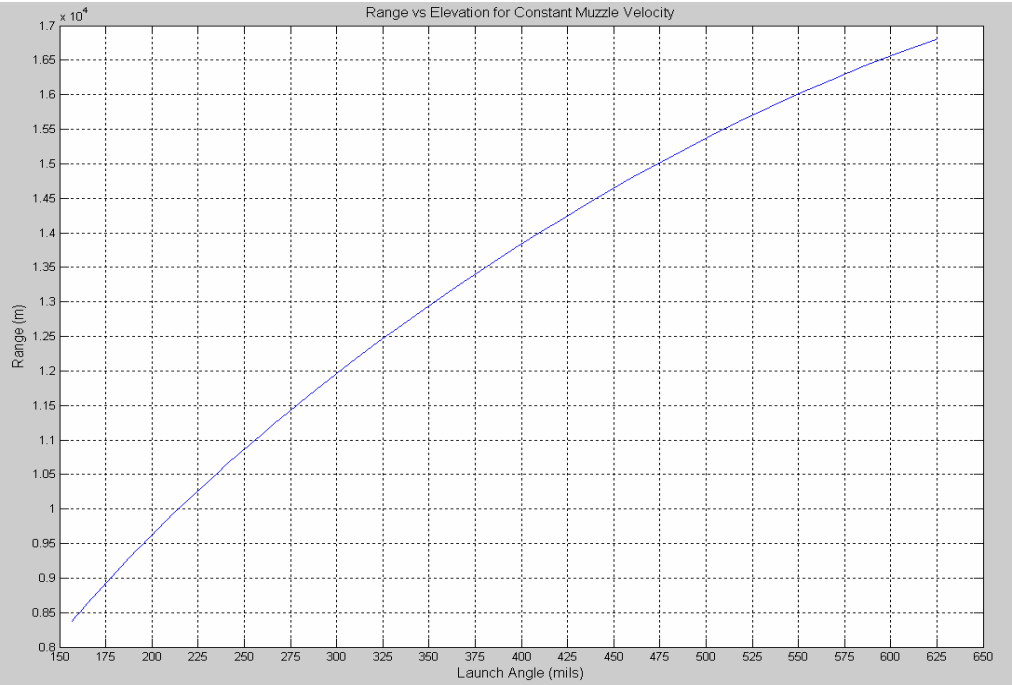

Figure 21. Range vs. Quadrant Elevation for Constant Muzzle Velocity

The variance in the muzzle velocity is also known as the muzzle velocity error budget. Error budgets of contributing factors are determined though measurements and tests. The error budgets can include other factors including the instrument errors and errors that may vary over time and distance, such as meteorological errors. For example, the meteorological conditions change with firing and target locations and the time lapsed after the last meteorological message, also known as the staleness hour.

The same analogy can be extended to other variations such as the launch angle and the wind effect. If each factor, such as the launch velocity and the launch angle, is independent, the total variation due to these two factors can be written as

$$\left(\sigma_X\right)^2 = \left(\sigma_{X,v}\right)^2 + \left(\sigma_{X,\theta}\right)^2, \tag{4.3}$$

where $\sigma_{X,v}$: Variation in range due to muzzle velocity, and

$\sigma_{X,\theta}$: Variation in range due to QE.

B PRECISION ERROR

The precision error is affected by the ballistic dispersion effect, and can be described as the round-to-round error within a single firing occasion. A single firing occasion contains multiple shots that are not affected by the time and spatial difference associated with MPI errors. Thus the PE is only dependent on the muzzle velocity, the form factor of the projectile and the quadrant elevation. The PE in a predicted fire in the range and deflection direction are given by,

$$\sigma_{PX}{}^2 = \left(\frac{\partial X}{\partial v}\sigma_v\right)^2 + \left(\frac{\partial X}{\partial Drag}\sigma_{Drag}\right)^2 + \left(\frac{\partial X}{\partial \theta}\sigma_\theta\right)^2, \tag{4.4}$$

$$\sigma_{PZ}{}^2 = \left(\frac{a_o}{1018.59} \times \frac{a_1 X}{a_1 - \theta}\right)^2, \tag{4.5}$$

where σ_{PX}: PE in range,

σ_{PZ}: PE in deflection,

$\frac{\partial X}{\partial v}$: Muzzle velocity unit effect,

$\frac{\partial X}{\partial Drag}$: Ballistic dispersion form factor unit effect,

$\frac{\partial X}{\partial \theta}$: QE unit effect,

σ_v: Error budget in muzzle velocity,

σ_{Drag}: Error budget in ballistic dispersion form factor,

σ_θ: Error budget in QE,

a_o: Constant, and

a_1: Constant.

The units of the PE errors in range and deflection are in meters. The equation used in the calculation of the PE in deflection represents an empirical fit to test data. It is dependent on the range to the target, *X*, and the quadrant elevation. From the PE deflection equation, an increase in range will lead to an increase in deflection. Similarly, an increase in the quadrant elevation will increase the deflection error.

1. Unit Effects in Precision Error

The unit effects that are used in the PE in range direction are related to the muzzle velocity, the launch angle, and the drag coefficient.

The **Muzzle Velocity Unit Effect** is calculated by varying the muzzle velocity 10 meters per second above the nominal muzzle velocity. The equation is given by

$$\frac{\partial X}{\partial v} = \frac{X_2 - X_1}{v_2 - v_1}, \tag{4.6}$$

$$v_2 = v_o + 10,$$

$$v_1 = v_o.$$

The ranges, X_2 and X_1 are calculated with the launch velocities of v_2 and v_1 respectively. Therefore, the muzzle velocity unit effect has the units of seconds.

The **Quadrant Elevation Unit Effect** is calculated by varying the launch angle 10 milradians above the nominal launch angle and is given by

$$\frac{\partial X}{\partial \theta} = \frac{X_2 - X_1}{\theta_2 - \theta_1}, \tag{4.7}$$

$$\theta_2 = \theta_o + 10,$$

$$\theta_1 = \theta_o.$$

The ranges, X_2 and X_1 are calculated with the launch elevation of θ_2 and θ_1 respectively. Therefore the quadrant elevation unit effect has the units of meters per milradians.

The drag on a projectile is dependent on the form factor. In addition, since the density varies linearly with the drag force, the **Ballistics Dispersion Form Factor Unit Effect** is calculated using the variation in the density (+10%) above the nominal density, also known as the **Density Unit Effect**. Therefore, the drag unit effect has the units of meters per percent. The density unit effect equation is given by

$$\frac{\partial X}{\partial Drag} = \frac{\partial X}{\partial \rho} = \frac{X_2 - X_1}{\rho_2 - \rho_1}, \tag{4.8}$$

$$\rho_2 = \rho \times 1.1,$$

$$\rho_1 = \rho .$$

2 Error Budgets in Precision Error

The error budgets used in this thesis for the calculation of the accuracy results were referenced to the error budgets used in JWAM such that the results can be compared in a later section.

The **Muzzle Velocity Error Budget**, σ_v, is dependent on the internal ballistics and the intermediate ballistics of the gun for a single firing occasion. Therefore it is a combination of the barrel and munitions conditions. The muzzle velocity standard deviation in the PE is due to the ballistics dispersion in a single occasion contrary to the MPI error which will be further explained in a later section. The referenced value from JWAM for σ_v is 1.99 meters per second.

The **Ballistics Dispersion Form Factor Error Budget**, σ_{Drag}, is dependent on the variation of the geometry of the projectile which would influence the aerodynamic drag. This form factor is used as a fitting factor used to match the prediction results with test data. The variation in the aerodynamic drag is measured in percent and the referenced value from JWAM is 0.65 percent.

The **Quadrant Elevation Error Budget**, σ_θ, is dependent on both the internal ballistics and intermediate ballistics of the gun for a single firing occasion. When the projectile is ejected, the barrel recoils and causes the gun to jump. When the projectile leaves the barrel, there will be a variation from the original barrel elevation. Even though

a gun jump experienced by a howitzer is appreciable, the effect on the departure angle of the projectile is not large since the duration of the projectile in the barrel is in the milliseconds region after the propellant is ignited. In addition, for PE, it is assumed for a single firing occasion the barrel returns to the same quadrant elevation after recoil. The referenced value from JWAM for σ_θ is 0.3 milradians.

C. MEAN POINT OF IMPACT ERROR

The MPI error is associated with the occasion-to-occasion variation about the target that is affected by the aiming error. In addition, the time and meteorological conditions difference affects the MPI error. For example, variation in meteorological conditions increases as time increases. The range and deflection MPI errors are given by,

$$\sigma_{MPI,X}{}^2 = \left[\frac{\partial X}{\partial \rho}\right]^2 \left(\sigma_\rho{}^2 + \sigma_{Drag}{}^2\right) + \left(\frac{\partial X}{\partial T}\sigma_T\right)^2 + \left(\frac{\partial X}{\partial w}\sigma_w\right)^2 + \left(\frac{\partial X}{\partial v}\sigma_v\right)^2 + \left(\frac{\partial X}{\partial \theta}\sigma_{AIM-EL}\right)^2 + \sigma_{LOC-X}{}^2 + \sigma_{CHART-X}{}^2, \quad (4.9)$$

$$\sigma_{MPI,Z}{}^2 = \left(\frac{\partial Z}{\partial w}\sigma_w\right)^2 + \left(\frac{\partial Z}{\partial (LIFT)}\sigma_{LIFT}\right)^2 + \left(\frac{\partial Z}{\partial \alpha}\sigma_{AIM-AZ}\right)^2 + \sigma_{LOC-Z}{}^2 + \sigma_{CHART-Z}{}^2, \quad (4.10)$$

where $\sigma_{MPI,X}$: MPI error in range,

$\sigma_{MPI,Z}$: MPI error in deflection,

σ_T: Error budget in temperature,

σ_w: Error budget in wind speed,

σ_{AIM-EL}: Error budget in aiming error for QE,

σ_{AIM-AZ}: Error budget in aiming error for azimuth,

σ_{LOC-X}: Error budget in location accuracy for range,

σ_{LOC-Z} : Error budget in location accuracy for deflection,

$\sigma_{CHART-X}$: Error budget in chart accuracy for range,

$\sigma_{CHART-Z}$: Error budget in chart accuracy for deflection,

σ_{LIFT} : Error budget in lift,

$\frac{\partial X}{\partial \rho}$: Density unit effect,

$\frac{\partial X}{\partial T}$: Temperature unit effect,

$\frac{\partial X}{\partial w}$: Range wind unit effect,

$\frac{\partial Z}{\partial w}$: Cross wind unit effect,

$\frac{\partial Z}{\partial (LIFT)}$: Lift unit effect, and

$\frac{\partial Z}{\partial \alpha}$: Azimuth unit effect.

1. Unit Effects in MPI Error

Since a unit effect measures the change in the range of deflection due to a change in the firing condition, the **Density Unit Effect**, **Muzzle Velocity Unit Effect**, and the **Quadrant Elevation Unit Effect** used in the MPI error are the same as in the PE and are re-used in the MPI error calculation.

The **Temperature Unit Effect** is calculated by varying the temperature by 10 percent above the temperature in standard condition. The temperature unit effect is given as

$$\frac{dX}{dT} = \frac{X_2 - X_1}{T_2 - T_1}, \tag{4.11}$$

$$T_2 = T_o \times 1.1,$$

$$T_1 = T_o.$$

The ranges, X_2 and X_1 are calculated with the launch elevations of T_2 and T_1 respectively, and keeping all other variables and inputs constant. The units of the temperature unit effect are meters per percent.

The **Wind Unit Effect** is separated into two components; the **Range Wing Unit Effect** and the **Crosswind Unit Effect**. The range wind unit effect is calculated by varying the wind speed in the range direction by 10 knots above the nominal wind speed resolved in the range direction. The units are given as meters per knots. So, the range wind unit effect is given as,

$$\frac{dX}{dw} = \frac{X_2 - X_1}{w_2 - w_1}, \tag{4.12}$$

$$w_2 = w_o + 10,$$

$$w_1 = w_o.$$

Similarly for the crosswind unit effect, the crosswind is varied by 10 knots above the nominal wind speed resolved in the deflection direction, and the units are given as meters per knots. The equations for the cross wind unit effect are

$$\frac{dZ}{dw} = \frac{Z_2 - Z_1}{w_2 - w_1}, \tag{4.13}$$

$$w_2 = w_o + 10,$$

$$w_1 = w_o.$$

The **Lift Unit Effect** is the change in deflection miss distance for a variation in the lift coefficient. In addition, the lift coefficient is dependent on the geometry of the projectile. It is also known that the drift of a projectile is dependent and according to the drift equation explained in an earlier section, the drift can be calculated using

$$Z_p = \frac{1}{2} a_{D,p} t^2.$$

The following equation can be used to calculate the drift in milradians,

$$Drift = Z_p \times \frac{X}{1018.59}, \tag{4.14}$$

where the value of 1018.59 is the conversion of radian to milradians (2π radians = 6400 milradians).

Assuming the relationship between the drift and lift is linear, i.e., a 1 percent increase in lift will lead to a 1 percent increase in drift, thus a 1 percent change in drift can be written as,

$$Drift(\%) = (Z_p \times \frac{X}{1018.59}) \times \frac{1}{100}. \tag{4.15}$$

Therefore, the lift unit effect, which has the units of meters per percent, is given as

$$\frac{\partial Z}{\partial (LIFT)} = \frac{Drift}{100} \times \frac{X}{1018.59}. \tag{4.16}$$

The **Azimuth Unit Effect** is the change in the deflection miss distance due to the change in the gun's azimuth. Assuming a small change in the azimuth angle, $d\alpha$, the change in deflection is given as

$$\partial Z = \frac{\partial \alpha}{1018.59} \times X. \tag{4.17}$$

where $d\alpha$ is in milradians. Therefore the azimuth unit effect with the units of meters per milradians, can be re-written as

$$\frac{\partial Z}{\partial \alpha} = \frac{X}{1018.59}. \tag{4.18}$$

2. Error Budgets in MPI Error

The **Density** (σ_ρ), **Temperature** (σ_T) and **Wind Error Budgets (**σ_W) are the variation of the meteorological conditions from the standard atmosphere. The variation is the cumulative effect on three independent factors; the instrument error, time difference and space difference. The error budgets for atmospheric conditions are not fixed as compared with other error budgets and are dependent on the staleness hour, which is the time lapsed after the last meteorological condition was measured. Table 3 shows the error budget for the meteorological condition with increasing staleness.

Met Message	0 Hours Staleness	1 Hour Staleness	2 Hours Staleness	4 Hours Staleness	No Met Message
σ_W (Knots)	0.8	4	4.9	7.2	11
σ_ρ (%)	0.15	0.4	0.69	0.97	6.6
σ_T (%)	0.25	0.3	0.57	0.79	3.0

Table 3. Error Budget Table for Meteorological Condition [see ref. 6]

The **Ballistics Dispersion Form Factor Error Budget**, σ_{Drag}, in the MPI error is similar to the error budget in the PE. This form factor is used in the occasion-to-occasion firing as a fitting factor used to match the prediction results with test data. The referenced value from JWAM is 1 percent.

Contrary to the **Muzzle Velocity Error Budget** in PE, the muzzle velocity error budget is dependent from the internal ballistics and the intermediate ballistics of the gun for occasion-to-occasion firing. Thus the muzzle velocity error budget in the MPI is the cumulative deviation due to the ballistics dispersion in ammunition and gun condition when firing from different occasions. The referenced value from JWAM for σ_v is 3 meters per second.

The **Aiming Error Budget (Quadrant Elevation),** $\sigma_{AIM\text{-}EL}$, is dependent on the aiming error caused by the mechanical system of the gun, elevation errors and instrumentation error from occasion-to-occasion firing. In addition, it also includes the gun jump effect mentioned in the quadrant elevation error budget in the PE error. The referenced value used is 0.5 milradians. Similarly, the **Aiming Error Budget (Azimuth)**,

$\sigma_{AIM\text{-}AZ}$, is dependent on the same factors other than the gun jump if it is assumed that the gun jump is only in the elevation plane. The referenced value from JWAM for $\sigma_{AIM\text{-}AZ}$ is 4 milradians.

The error budget for the gun and target location, **Range Location Error Budget** ($\sigma_{LOC\text{-}X}$), and **Deflection Location Error Budget** ($\sigma_{LOC\text{-}Z}$) depends on the accuracy of the instrument determining the gun firing position. Global Positioning Systems (GPS) and Inertial Navigation Systems (INS) are typically used in determining the location. The referenced value from JWAM used for both $\sigma_{LOC\text{-}X}$ and $\sigma_{LOC\text{-}Z}$ is 15 meters.

If the gun and target positions are obtained from a map or chart, the **Chart Accuracy in Range Error Budget** ($\sigma_{CHART\text{-}X}$) and **Chart Accuracy in Deflection Error Budget** ($\sigma_{CHART\text{-}Z}$), the standard deviation is dependent on how accurate the map is read. However, modern fire control system using digitalized maps effectively reduce the errors for chart accuracy to zero.

THIS PAGE INTENTIONALLY LEFT BLANK

V. RESULTS AND DISCUSSION

A. TRAJECTORY RESULTS

It is critical to develop an accurate trajectory model because the unit effects are calculated from the trajectory model. If the trajectory model is inaccurate, this would subsequently lead to errors in the unit effects and accuracy equations.

Table 4 shows the comparison of the results from the 3 DOF model and the NABK program, which is a modified point mass model. The inputs are as follow:

1. Projectile Mass: 42 kilograms
2. Muzzle Velocity: 684.5 meters per second
3. Standard Meteorological Conditions
4. Range: 5,000 meters, 10,000 meters , 15,000 meters, Max Range

In addition, in Table 4, the trajectory results using a zero drag model are also calculated using the muzzle velocity of 684.5 meters per second and the quadrant elevation calculated from the 3 DOF model. Therefore, the range and the outputs—quadrant elevation, terminal velocity, angle of fall (impact angle), and time of flight—can be compared.

			Trajectory				
S/N	Program	Low or High Angle Fire	Range (m)	QE (mils)	Terminal Velocity (m/s)	Angle of Fall (mils)	Time of flight (s)
1	NABK	Low	5000	69.587	441.151	103.420	9
	3 DOF	Low	5031.077	72.97	440.933	97.421	9.2
	Zero Drag	Low	6812.171	72.97	684.5	72.97	9.9776
2	NABK	Low	10000	204.079	312.516	365.422	23.354
	3 DOF	Low	10038.57	215.872	309.1	370.951	23.95
	Zero Drag	Low	19633.606	215.872	684.5	215.872	29.3435
3	NABK	Low	15000	446.879	314.331	713.177	43.305
	3 DOF	Low	15000.43	473.184	312.084	731.217	44.44
	Zero Drag	Low	38261.097	473.184	684.5	473.184	62.5225
4	NABK	Low	18246.6	788.011	328.182	1035.390	67.689
	3 DOF	Low	17702.87	776.463	324.73	1017.009	65.62
	Zero Drag	Low	47710.494	776.463	684.5	776.463	96.3714
5	NABK	High	18246.6	846.859	329.859	1082.037	71.784
	3 DOF	High	17700.81	860.87	327.729	1083.840	71.03
	Zero Drag	Low	47420.761	860.87	684.5	860.87	104.3591
6	NABK	High	15000	1129.23	335.749	1293.052	88.803
	3 DOF	High	15039.26	1117.297	336.429	1267.218	85.04
	Zero Drag	Low	38796.048	1117.297	684.5	1117.297	124.1639

Table 4. Comparison of Trajectory Results from NABK, 3 DOF, and Zero Drag Model

1. Discussion of the Trajectory Results

The comparison between the zero drag model, the 3 DOF model, and the NABK model can be seen from by comparing Figure 22 to Figure 25. In Figure 22, the range predicted by the zero drag model differs from that of both the 3 DOF and NABK. The differences increase as the range increases. In Figure 23, the zero drag model has a constant terminal velocity, regardless of the ranges, which is due to exclusion of the drag force. In Figure 24, the zero drag model predicts a angle of fall that is always lower than that of both the 3 DOF and the NABK. The angle of fall in a zero drag model is exactly the same as the quadrant elevation during launch. In addition, both the 3 DOF and NABK models predict a steeper angle of fall due to the resistance to the drag force. Finally, in Figure 25, the differences in the prediction of the time of flight using the zero drag increase as the range increases. This is because the time of flight is related to the range

and is dependent on the drag force. Therefore, for an artillery projectile, the consideration of drag force in the trajectory model is very important.

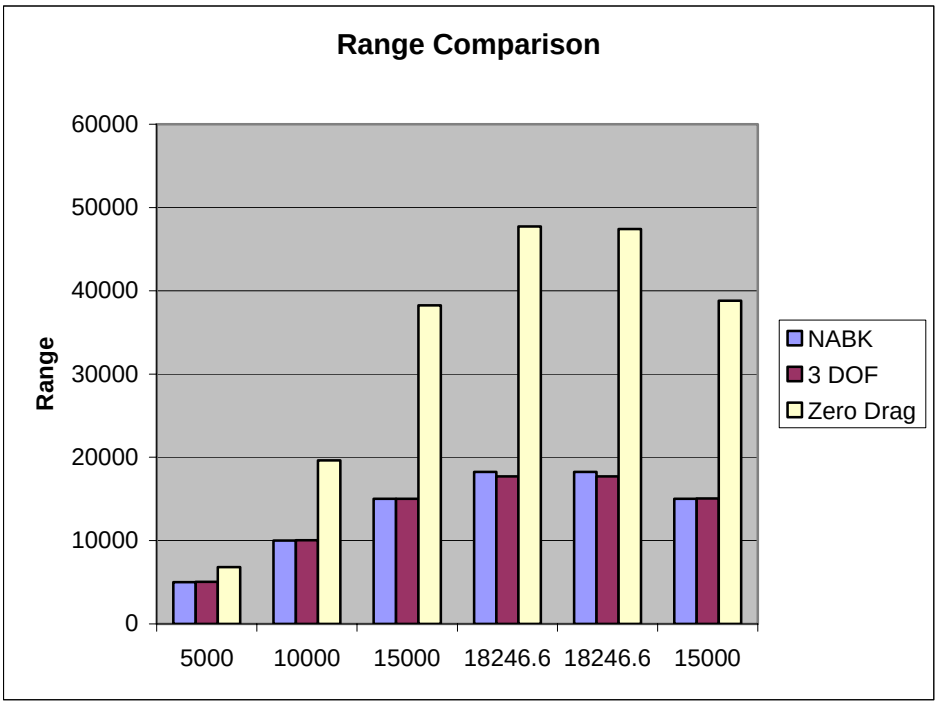

Figure 22. Range Comparison Between the Three Models

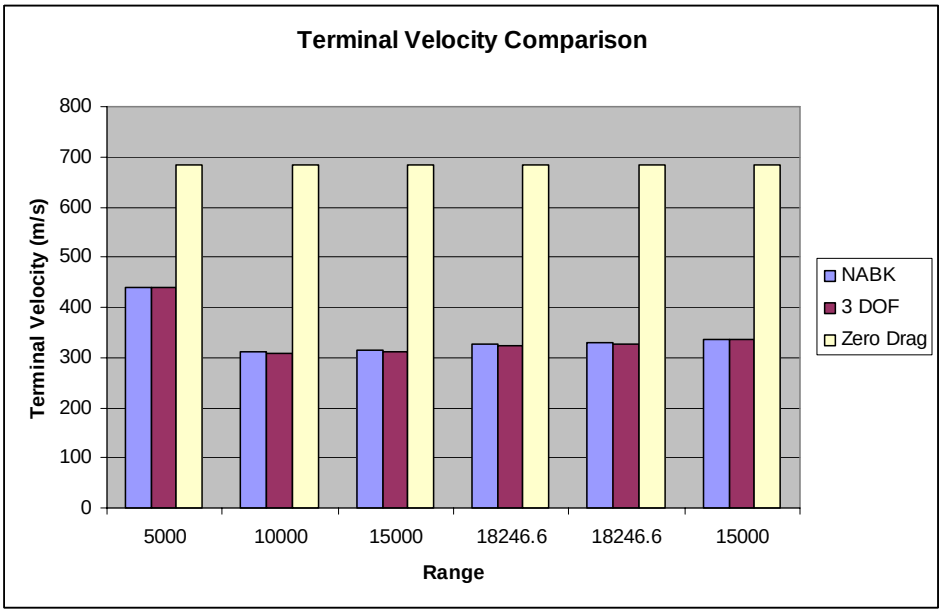

Figure 23. Terminal Velocity Comparison Between the Three Models

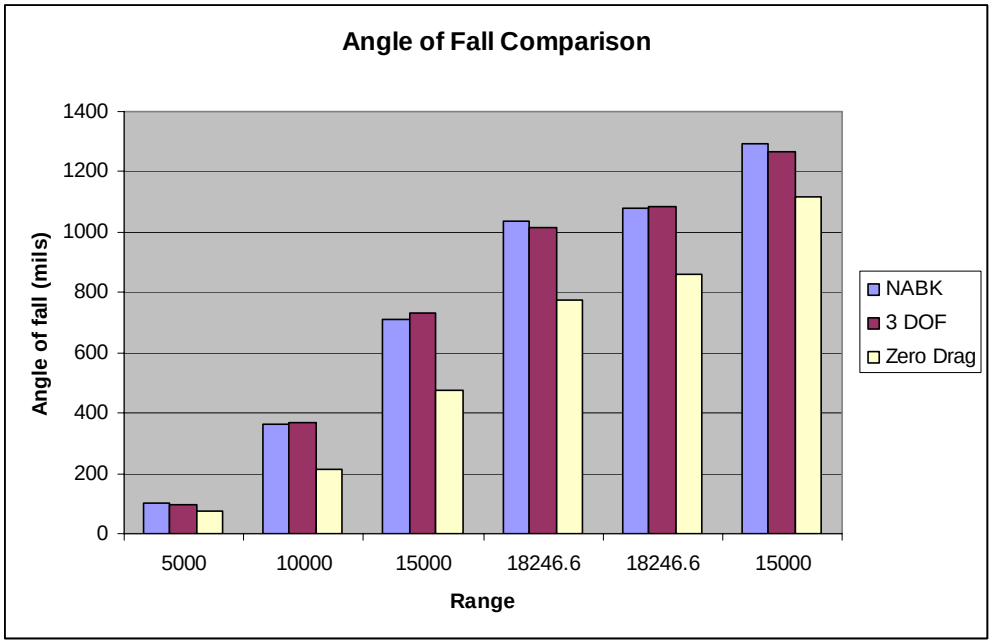

Figure 24. Angle of Fall comparison Between the Three Models

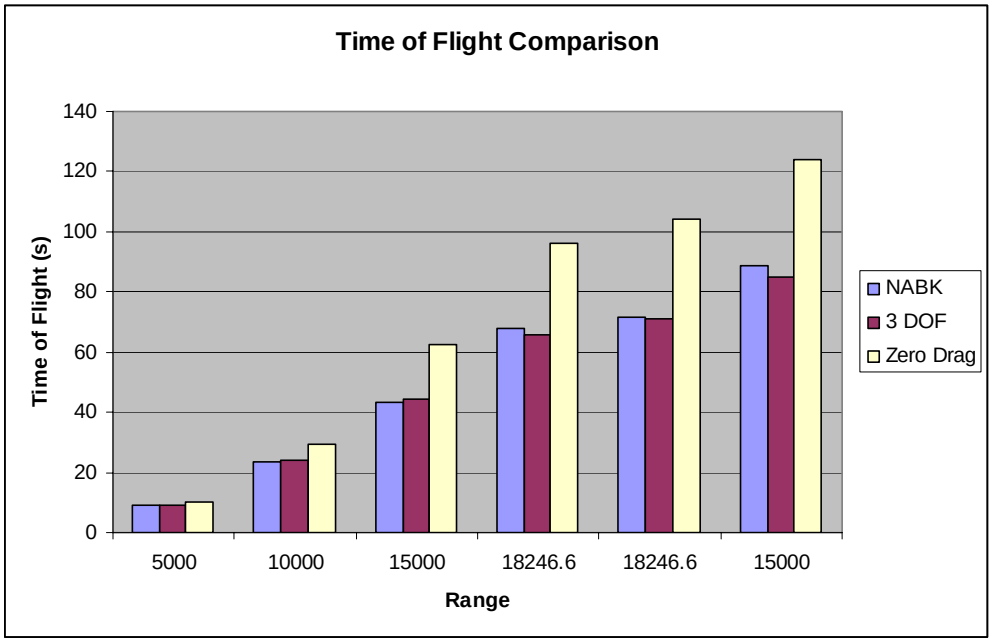

Figure 25. Time of flight comparison Between The Three Models

In the 3 DOF model, since the range is an input, the trajectory program will perform a incremental increase in quadrant elevation, $\Delta\theta$, from 5 milradians until it reaches the desired range. As a result, the output range according to the quadrant elevation will be slightly different from the input range and is dependent on the incremental increase in quadrant elevation. As seen in Table 4, the output range value from the 3 DOF model differs from the NABK model.

Based on the inputs to the trajectory model, the NABK is able to predict a max range of 18,246.6 meters, as compared to an estimate of 17,702.87 meters from the 3 DOF model. The difference could be due to the exclusion of the Magnus effect in the 3 DOF model since the Magnus effect generates lift and can increase the range of the trajectory.

Figure 26 shows the comparison of the range, terminal velocity, angle of fall, and time of flight with quadrant elevation. From the trajectory results in Table 4, it can be seen that the 3 DOF trajectory model is able to predict a trajectory relatively accurate when compared with the NABK model. Using a simple 3 DOF model is sufficient to show the general behavior of a projectile launched from a howitzer.

It is expected that as the quadrant elevation increases, the differences between the 3 DOF model and the NABK program would increase due to the exclusion of the Coriolis and Magnus effects in the 3 DOF model.

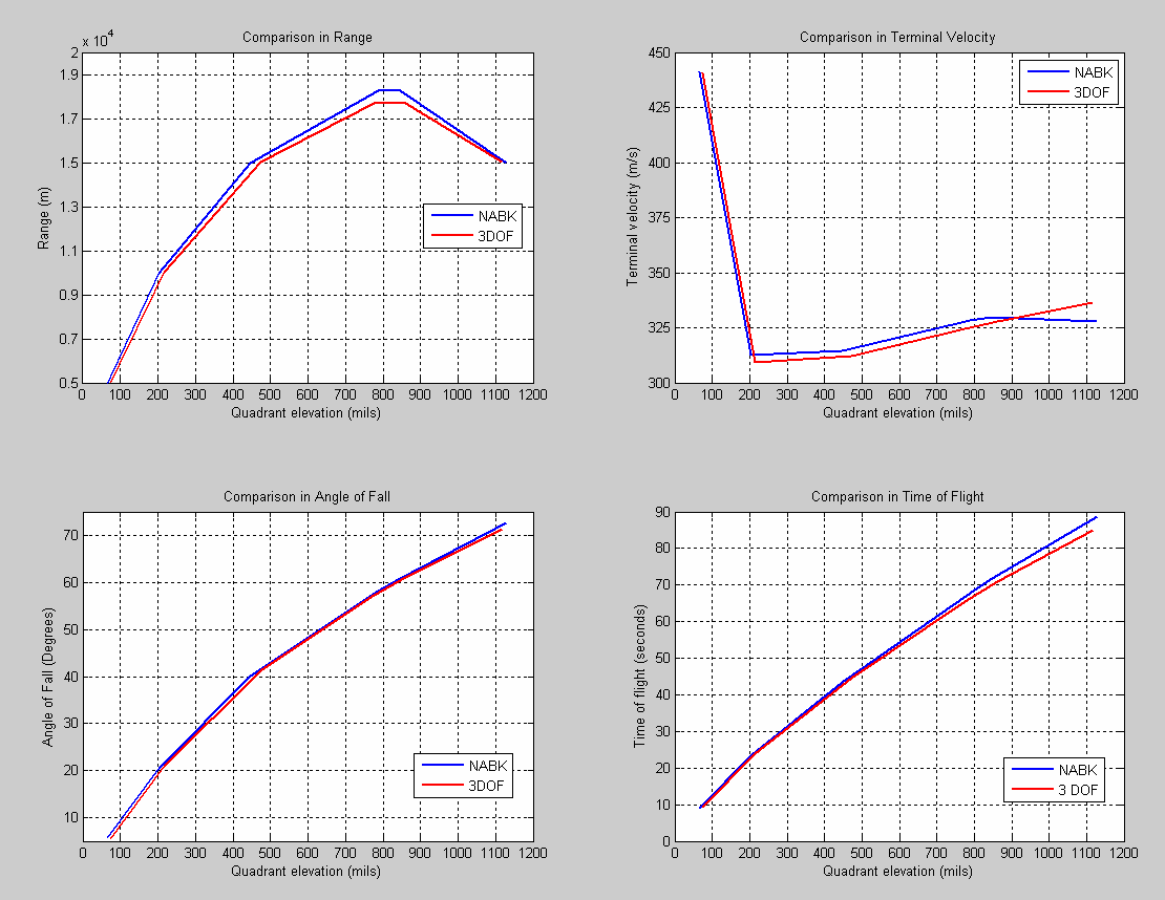

Figure 26. Comparison of Output Results with Quadrant Elevation between NABK and the 3 DOF Model

B. ACCURACY RESULTS

The accuracy results are compared between the thesis accuracy model (TAM) and the JWAM. The JWAM gets its inputs from the NABK program. In addition, the JWAM program is able to calculate the unit effects or retrieved stored values of the unit effects. The unit effects are also calculated in the TAM.

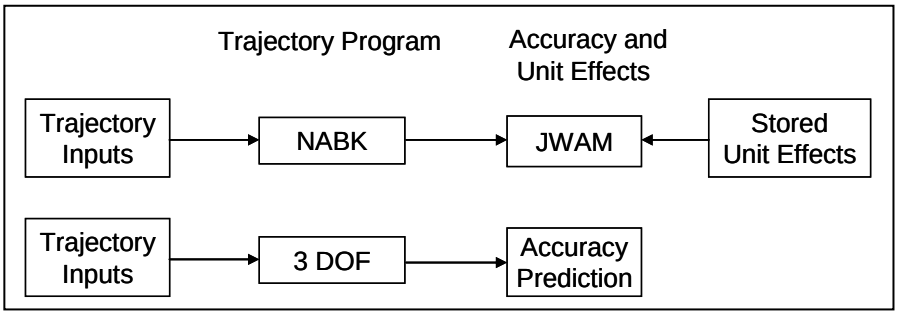

Figure 27. Accuracy Model in NABK and Accuracy Model in the Thesis

Table 5 displays the accuracy results for range and deflection in PE and MPI error in predicted firing calculated from the TAM and JWAM at ranges of 5,000 meters, 10,000 meters, and 15,000 meters. Table 6 shows the unit effects calculated by the TAM and the JWAM at maximum range.

		PE (m)		MPI Error (m)	
Program	**Range**	**Range Error**	**Deflection Error**	**Range Error**	**Deflection Error**
JWAM	5000	29.5436	3.92077	52.2662	25.1544
TAM	5031.077	28.999	3.952	51.944	25.471
JWAM	10000	47.6412	8.42877	89.2109	46.4748
TAM	10038.57	46.594	8.517	89.815	50.348
JWAM	15000	63.15	14.6197	130.239	77.6191
TAM	15000.43	62.331	14.8777	138.937	89.624
JWAM	18246.6	78.4241	22.7895	182.537	110.761
TAM	17702.87	75.52	21.902	178.095	114.885

Table 5. Range and Deflection Error for PE and MPI

Program	Range	Temp Partials (m/%)	Range Wind Partials (m / m/s)	QE Partials (m/degree)	Density Partials (m/%)	MV Partials (s)	Cross wind Partials (m/kts)	Deflection (mils)
JWAM	5000	3.1739	2.6654	923.7435	12.121	11.9737	1.719	-2.33
TAM	5031.077	-2.011	3.542	886.945	-12.321	11.815	2.113	-1.738
JWAM	10000	7.437	12.6886	472.827	41.4151	19.3408	8.5357	-7.073
TAM	10038.57	-5.061	14.854	444.312	-41.811	18.642	11.849	-5.905
JWAM	15000	9.3582	32.0815	280.067	63.8911	23.7881	19.241	-15.437
TAM	15000.43	11.533	37.857	257.32	-64.7	23.015	26.279	-13.605
JWAM	18246.6	17.8221	49.7887	24.7289	85.6366	27.7599	30.748	-32.93
TAM	17702.87	18.476	49.137	34.56	-83.633	26.331	33.498	-25.135

Table 6. Comparison of the Unit Effects between the JWAM and TAM

1. Discussion of Accuracy Results

a. Comparison of Accuracy Results between the JWAM and TAM

Table 5 shows the accuracy results for range and deflection in PE and MPI error in predicted firing calculated from the TAM and JWAM. The results from both the TAM and the JWAM for PE for all ranges are very close with the maximum difference of 2.9401 meters at maximum range The prediction of the deflection error in PE from the TAM was expected to be accurate when compared with the JWAM since the equation is empirical and contains only the constants a_0 and a_1 and is dependent on the range and quadrant elevation. However, in the MPI error in the 15,000 meter range, there is a difference of 8.698 meters and 12 meters in the range and deflection, respectively. Since the error budgets used are the same for both programs, it is suspected that the discrepancies lie with the unit effects.

b. Discussion in Unit Effects

From Table 6, it is noticed that there are negative values in the temperature unit effects and the density unit effects. The negative sign corresponds to the fact that an increase in the temperature or density in that quadrant elevation would bring about a decrease in the range. However, this does not affect the accuracy calculation as the PE and MPI error is the sum of square of the individual terms.

From the results, the TAM is able to predict the unit effects that are close to the JWAM values. In the **Deflection** column, Table 5 shows the drift caused by the spinning projectile effect. Differences in the results in expected due to the fact that the trajectory program in the thesis uses an average weighted acceleration as compared with the NABK which uses the equations of motion of the projectile due to the gyroscopic effect.

Figure 21 shows that the unit effect is not constant and it varies with range. At every launch angle, the unit effect is different. The variation at which the unit effect is calculated affects the results of the unit effects. For instance, in the QE unit effect, using a variation of ± 10 milradians and 0 to 10 milradians gives a different unit effect. Intuitively, a ±1 milradians about the nominal quadrant elevation would generate

the most accurate unit effect. This is illustrated in Figure 28. It is desirable to investigate the effect of the variation used in the calculation of the unit effect.

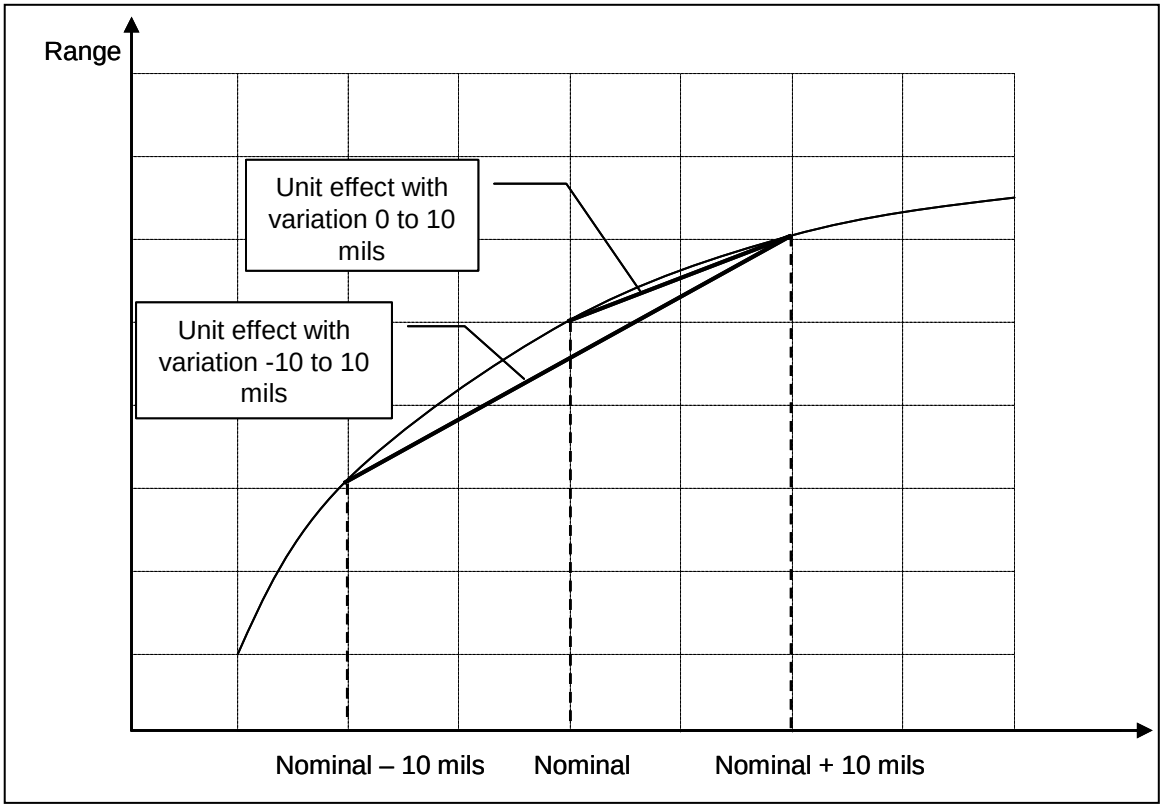

Figure 28. Effects of Variation on the Unit Effect

Table 7 shows the results of the unit effects when calculated with different variations with the TAM in comparison with the unit effects from JWAM at a range of 15,000 meters. In addition, it also shows how the unit effect influences the accuracy results at different variations. Table 8 shows the same comparison but at a range of 10,000 meters.

Table 7 shows that the variation in the unit effects did not contribute much to the differences in the accuracy result. In fact, using the same variation as JWAM, results in the calculated MPI errors are closer to the accuracy results calculated by JWAM. The unit effects shown in both Table 7 and Table 8 closely resemble the unit

effects form JWAM. The exception was the unit effect for the cross wind. It was explained in an earlier section that this is caused by the two different trajectory models.

In Table 8, the ±1 variation for the temperature and range wind unit effect generates a rounding off error when the respective unit effects were calculated. However, even though there is a rounding off error, the effect on the accuracy results is almost negligible. This error's influence on the accuracy results is also dependent on the error budgets. For example, if the error budget in the temperature is big and when the accuracy result is calculated using the ±1 % variation, the contribution to the accuracy result would be large compared with when the error budget of the temperature is small. Therefore, the accuracy results do not depend only on the unit effects, but are dependent on the error budget too.

Models	Variation	Temp Unit Effect (m/%)	Density Unit Effect (m/%)	Muzzle Velocity Unit Effect (s)	Range Wind Unit Effect (m/m/s)	Cross wind Unit Effect (m/kts)	QE Unit Effect (m/deg)
TAM	±10	10.397	-69.279	22.989	37.497	26.279	259.687
TAM	±5	10.123	-69.211	22.989	37.722	26.279	259.694
TAM	±1	10.032	-69.685	22.048	39.55	26.279	242.982
TAM	0 to 5	10.922	-66.951	23.002	37.789	26.279	258.511
TAM	0 to 10	11.533	-64.7	23.015	37.857	26.279	257.32
JWAM	0 to 10	9.3582	63.8911	23.7881	32.0815	19.241	280.067

Models	Variation	PE		MPI	
		Range Error	Deflection Error	Range Error	Deflection Error
TAM	±10	64.342	14.878	141.011	89.624
TAM	±5	64.311	14.878	141.339	89.624
TAM	±1	63.195	14.878	143.378	89.624
TAM	0 to 5	63.31	14.878	140.12	89.624
TAM	0 to 10	62.331	14.8777	138.937	89.624
JWAM	0 to 10	63.15	14.6197	130.239	77.6191

Table 7. Unit Effects with Different Variations for Range of 15, 000 Meters

Models	Variation	Temp Unit Effect (m/%)	Density Unit Effect (m/%)	Muzzle Velocity Unit Effect (s)	Range Wind Unit Effect (m / m/s)	Cross wind Unit Effect (m/kts)	QE Unit effect (m/deg)
TAM	±10	-6.439	-44.624	18.756	15.0952	11.849	452.35
TAM	±5	-6.6335	-44.629	18.9	15.38	11.849	452.245
TAM	±1	-42.3182	-45.449	19.478	45.0733	11.849	452.295
TAM	0 to 5	-5.9421	-43.004	18.915	15.403	11.849	450.795
TAM	0 to 10	-5.061	-41.811	18.642	14.854	11.849	444.312
JWAM	0 to 10	7.437	41.4151	19.3408	12.6886	8.5357	472.827

Models	Variation	PE		MPI	
		Range Error	Deflection Error	Range Error	Deflection Error
TAM	±10	47.882	8.517	96.68	50.343
TAM	±5	48.108	8.517	93.383	50.343
TAM	±1	49.329	8.517	93.367	50.343
TAM	0 to 5	47.498	8.517	91.928	50.343
TAM	0 to 10	46.594	8.517	89.81	50.343
JWAM	0 to 10	47.6412	8.42877	89.2109	46.4748

Table 8. Unit Effects with Different Variations for Range of 10, 000 Meters

c. Error Terms

The error term is the multiplication of the unit effect and the respective error budget as explained in Chapter IV. Therefore, from the error terms, it can be readily shown what the major contributors to the accuracy results are. Table 9 shows the error terms for the PE in the range direction.

In Table 9 it can be seen that the Muzzle Velocity has a big influence on the accuracy result. A deviation in the error budget of the muzzle velocity would have a greater effect as compared with the other two error terms. In addition, for the QE, the error term reduces as the range increases to maximum range. This is due to the fact that at low QE, a small deviation in the QE will cause a big change in range compared with the same deviation at QE nearer to 45 degrees.

Program	Range (m)	Muzzle Velocity (m)	Form Factor (m)	QE (m)	Range Error (m)
JWAM	5000	23.8276	7.8786	15.5881	29.5436
TAM	5000	23.5118	-8.0086	14.9671	28.999
JWAM	10000	38.4881	26.9198	7.9789	47.6412
TAM	10000	37.0975	-27.1771	7.4977	46.594
JWAM	15000	47.3383	41.5292	4.7261	63.15
TAM	15000	45.7998	-42.055	4.3422	62.331
JWAM	18246.6	55.2422	55.6637	0.4173	78.4241
TAM	17702.87	52.3986	-54.3614	0.5832	75.52

Table 9. Error Terms for PE in Range

Table 10 shows the constants for calculating the PE error in deflection. The deflection error is calculated using equation (4.5) and it is dependent on the range and QE. Since a_0 and a_1 are constant, an increase in either the range or the QE will lead to an increase in the deflection error

Program	Range (m)	QE (mils)	a_0	a_1	Deflection Error (m)
JWAM	5000	69.587	0.52	2000	3.92077
TAM	5000	72.97	0.52	2000	3.952
JWAM	10000	204.079	0.52	2000	8.42877
TAM	10000	215.872	0.52	2000	8.517
JWAM	15000	446.879	0.52	2000	14.6197
TAM	15000	473.184	0.52	2000	14.8777
JWAM	18246.6	788.011	0.52	2000	22.7895
TAM	17702.87	776.463	0.52	2000	21.902

Table 10. Error Terms for Deflection Error in PE

Table 11 shows the error terms for MPI error in the range direction. The major contributors to the accuracy result are the error terms due to the deviation in form factor, range wind, and muzzle velocity. The influence of the range wind increases rapidly as the range increase. This is due to the longer flight time in longer ranges leading to the projectile being exposed to the wind effect. In cases where the wind speed is low,

the influence of the range wind will be small. In addition, where regular updates of meteorological conditions are available, the error budget for the wind would be smaller, leading to improved accuracy. It is evident that all of the error terms, other than the deviation caused by the QE, increase as the range increases. The reduction in the QE error term is the same as explained earlier for the PE.

Program	Range (m)	Form factor (Density & Drag) (m)	Temp (m)	Range Wind (m)	Muzzle Velocity (m)	QE (mils)	Location (m)	Chart (m)	Range Error (m)
JWAM	5000	21.9523	2.5708	7.29	35.92	25.99	15	0	52.27
Thesis	5000	22.3146	-1.6289	9.69	35.45	24.95	15	0	51.94
JWAM	10000	57.4103	4.9084	29.44	58.02	13.30	15	0	89.21
Thesis	10000	57.9591	-3.3402	34.46	55.93	12.50	15	0	89.82
JWAM	15000	71.4324	4.6791	80.36	71.36	7.89	15	0	130.24
Thesis	15000	72.3367	5.7665	94.83	69.05	7.24	15	0	138.94
JWAM	18246.6	89.9143	7.4852	133.95	83.28	0.70	15	0	182.54
Thesis	17702.87	87.8106	7.7599	132.19	78.99	0.97	15	0	178.10

Table 11. Error Terms for MPI in Range

Similarly, in Table 12, the crosswind effect on the accuracy increases with the time of flight. In addition, the error term for the deviation in the gun aiming in the azimuth plane is a major contributor to the accuracy results. Evidently, a small deviation in the azimuth leads to an increase in the deflection error as the range increase.

Program	Range (m)	Cross Wind (m)	Form Factor (Drift) (m)	Azimuth (m)	Location (m)	Chart (m)	Deflection Error (m)
JWAM	5000	4.7042	-0.1143	19.634	15	0	25.1544
Thesis	5000	5.7824	-0.0858	19.7570	15	0	25.471
JWAM	10000	19.8019	-0.6943	39.2699	15	0	46.4748
Thesis	10000	27.4884	-0.5819	39.4214	15	0	50.348
JWAM	15000	48.1987	-2.2732	58.9049	15	0	77.6191
Thesis	15000	65.8288	-2.0035	58.9066	15	0	89.624
JWAM	18246.6	82.7213	-5.8989	71.6543	15	0	110.761
Thesis	17702.87	90.1196	-4.3684	69.5191	15	0	114.885

Table 12. Error Terms for MPI in Deflection

From the results, it is evident that if the deviations in the major contributing error terms, such as deviations in muzzle velocity, form factor, and azimuth aiming errors, could be controlled, the accuracy result will improve. Similarly, when firing in a zone where the deviations in the wind speed are small, the accuracy will improve.

VI. CONCLUSION

From the discussion of the trajectory and accuracy results, the following can be concluded:

1. Consideration of the drag force in trajectory calculation is important in projectiles fired from a howitzer. The zero drag model predicts a trajectory that is inaccurate in practical cases.

2. A 3 DOF model is sufficient to show the general behavior of the trajectory of an artillery-fired projectile. However, it cannot predict the drift as accurately as the modified point mass model. In addition, as the time of flight increases, the discrepancies in results between a 3 DOF and modified point mass model increase due to the exclusion of the Coriolis and Magnus effects in the 3 DOF model. This is the result of the effects of the Coriolis and Magnus acceleration at high QE which can cause considerable drift.

3. A 3 DOF trajectory model is easy to implement and the computation is less intensive that the NABK model, which is a 5 DOF model. The simplicity of the 3 DOF model enables greater insight into the mechanics of the trajectory, which the 5 DOF does not, while still producing accurate results.

4. The TAM is able to predict the accuracy result, compared with the JWAM program, in the range of 0.6% to 15.5% where in most cases, the errors are less than 4%.

5. The unit effect varies with range. The variations used in the calculation of the unit effects do not strongly influence the accuracy results. However, when the variation is small, rounding error in the calculation of the unit effects might occur.

6. In PE for range, the deviation in the muzzle velocity is the major contributor to the accuracy results.

7. Deflection error in PE can be accurately calculated using the TAM as the empirical formula is only dependent on the range and QE.

8. In MPI error for range, the major contributors to the accuracy results are the muzzle velocity and the range wind.

9. In the MPI error for deflection, the major contributors to the accuracy result are the cross wind effect and the gun aiming in the azimuth plane.

10. Reducing the deviation in the error budget reduces the accuracy errors. For instance, if the muzzle velocity can be better controlled, the accuracy error will reduce. This is similar for meteorological conditions. The error budgets for wind, density, and temperature will reduce if the staleness hour is small.

11. This thesis showed the methodology in calculating the trajectory and showed how the PE and MPI errors can be calculated. The methodologies are the same for predicted and adjusted fire since the errors are the root sum square of all the related error terms. Since calculating accuracy in predicted fire is simpler due to fewer contributing factors, understanding the methodology in predicted fire will effectively aid in the modeling of accuracy prediction in adjusted fire.

APPENDIX I. SOFTWARE IMPLEMENTATION

The software models for the 3 DOF trajectory model and the TAM was written using Matlab ver 7.1 and is separated into different modules, the **Input**, **Main Program**, **Met Data**, **Trajectory Program**, **Acoustic Data** and **Drag Coeff** M-files.

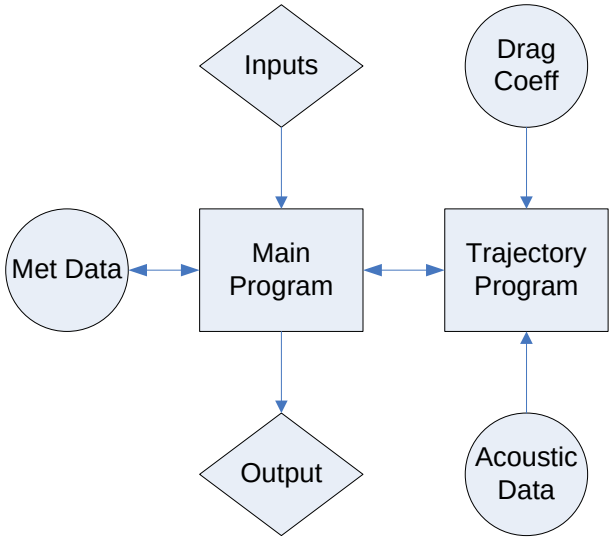

Figure 29. Modular files in the software model

A. FUNCTIONS OF THE SOFTWARE

The inputs used to calculate the nominal trajectory and the partial derivatives in the program can be found in the **Input** file and is written using ASCII format shown in Figure 31. It allows the user to input the desired range, muzzle velocity and meteorological conditions such that the nominal trajectory can be predicted. In addition, the input file contains the error budgets for the calculation of the PE and the MPI errors.

The **Main Program** reads the inputs from the input file. Variables and constants that do not change such as the gravitational force and gas constant for the calculation of the trajectory are hard-coded in this file. Other than the trajectory calculation, all calculations such as the errors and the unit effects are performed in this program. The results are compiled and displayed in an output file.

The sole purpose of the **Trajectory** M-file is to calculate the trajectory of the projectile given the inputs from the main program. The trajectory program can be used in any other Matlab programs provided the input requirements are met. The main program controls the variables to be passed to the trajectory program instead of reading directly from the input file. For instance, in the calculation of the nominal trajectory, the inputs to the trajectory program are the same as from the input file; however, in calculating the unit effect for muzzle velocity, the muzzle velocity that would be passed to the trajectory program would be different from the nominal value read from the input file.

The **Drag Coeff** and **Acoustic Data** M-file contain the drag coefficients of the projectile and the standard meteorological data. The data are then passed to the trajectory program. The **Met** M-file contains the standard deviations of the wind speed, temperature and density for different staleness times.

The flow chart of the software is as shown in Figure 30. When the inputs and the meteorological data are known, the software will determine whether the range is within the maximum and minimum range for the specified muzzle velocity. Next, the software will calculate the required QE by incrementally increase the launch angle. With the required QE, the software will output the trajectory results if no error calculations are required shown in Figure 32. On the other hand, if the error calculations are required, the software would calculate the unit effects. With the error budgets specified in the inputs, the software would then be able to calculate the accuracy results shown in Figure 33.

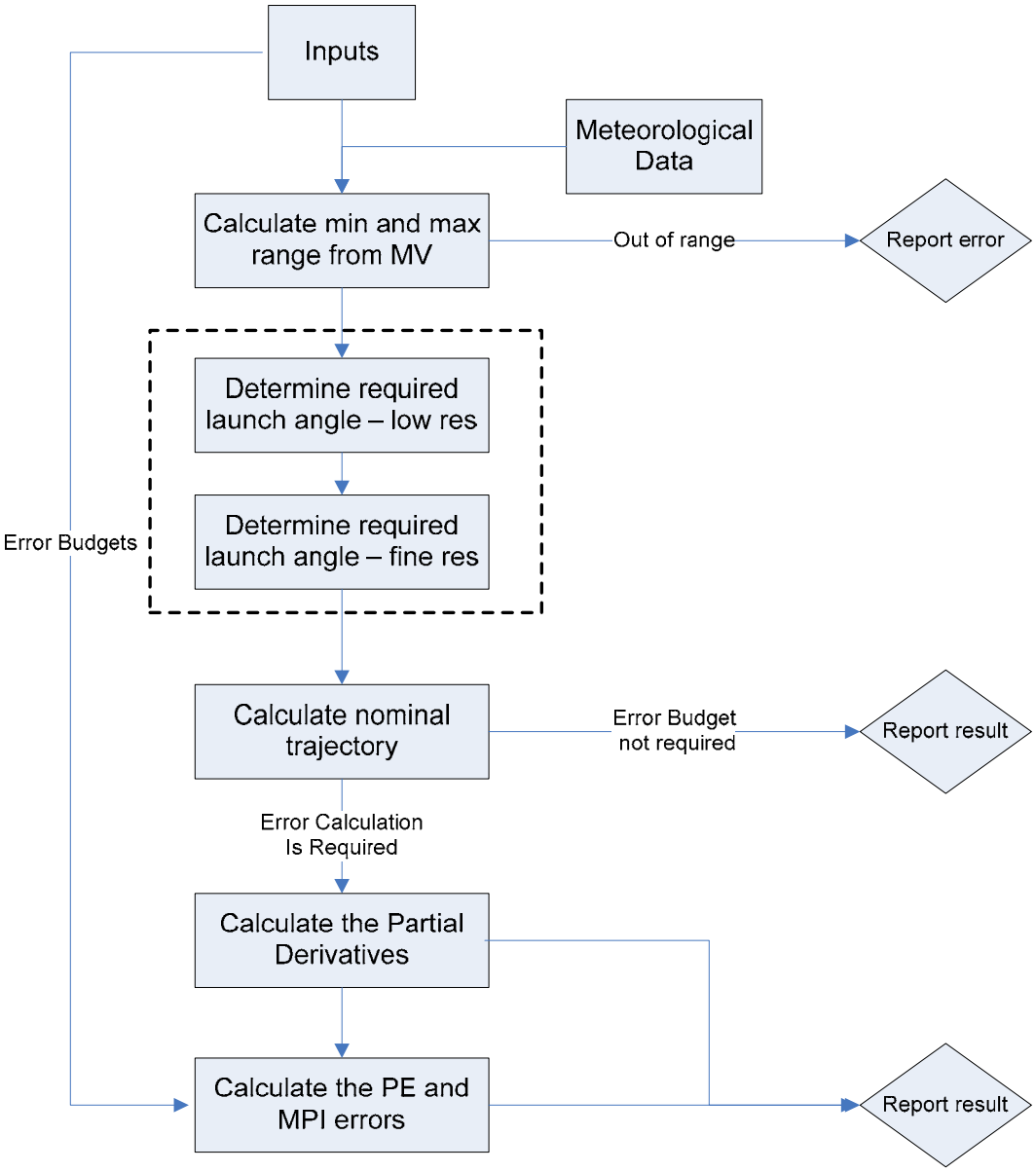

Figure 30. Software Flow Chart

B. IMPROVEMENTS TO THE PROGRAM

The software model used in the trajectory prediction is not optimized. This is due to the fact that the program has to incrementally increases the QE with a given muzzle velocity such that it meets the range. Thus the program has to perform the 3 DOF trajectory model at every QE incremental angle unit it reach the desired range. This is computationally intensive. In addition, the accuracy of the trajectory is dependent on how small the incremental steps are.

The apex height must be known such that the correct line of the meteorological message can be entered into the input file. The present program assumes that a user knows the apex height of the trajectory. The program can be modified such that the user does not need to enter the apex height, but the program would be able to read in a typical meteorological message and using the trajectory program to predict the height and thus selecting the correct line in the meteorological message.

A constant time step of 0.01 seconds is used in the program. This time step can be changed depending on the range. For instance, when calculating the trajectory with a range of 5,000 meters, a smaller time step can be used. For large ranges, the time step can be larger. The effect of the time step on the accuracy of the trajectory calculation was not optimized. If it is found that the time step can be increased, it will reduce the computational time.

The command “fscanf” in Matlab is used to read the input file. It reads in every line even though the description is not used in the program. Thus computational resources are wasted. In addition the spacing and indents are critical. If a user accidentally deleted a word or space from the input file, the program would read in the wrong data. The command can be further optimized to just read in the required data and improving it to be insensitive to any accidental changes in the input file.

C. SAMPLE INPUT /OUTPUT FILE

A sample of the input file is is as shown in Figure 31. All the inputs to the program are entered in this file such that there will be no other required actions when the program runs. Other than the inputs for the trajectory calculation, the error budgets are also entered in this file. The exception is the error budget for the meteorological conditions which resides in the **Met** M-file. However, the error budget is dependent on the staleness time which can be specified in the input file. The program mode allows the user the option to use the program just as a basic trajectory program or to calculate both the trajectory and accuracy results.

```
----Input file for the trajectory program and error margins calculation----

----------------Program Mode-----------------------------------------------

1               Input 1 to calculate the error margin. Input 0 for trajectory

----------------Inputs for the projectile----------------------------------

42              Mass of Projectile (Kg)
0.155           Diameter of the projectile (m)
0.2029          Deflection acceleration (m/s^2)

----------------Inputs for the desired range and muzzle velocity-----------

15000           Desired range (m)
684.5           Muzzle velocity (m/s)
0               Direction of target (deg). 0-north / 90-east
1               Input 0 for low elevation and 1 for high elevation

----------------Inputs for the Met message---------------------------------

0               Direction of wind (deg). 0 - towards north / 90 - towards east
0               Wind speed (knots)
1               Air temp variation from std atm (1.0 - std atm)
1               Air density variation from std atm (1.0 - std atm)
2               Input 0, 1, 2, 4 for staleness hour and 5 for no met data

----------------Inputs for standard deviations for Precision Error---------

1.99            Muzzle velocity (m/s). Reference: BRL
0.65            Drag /Adjusment factor for precision error (%)
0.016875        Barrel elevation. Based on 0.3 mils accuracy. (deg)
0.52            Constant for deflection calculation, a_0
2000            Constant for deflection calculation, a_1

----------------Inputs for standard deviations for MPI Error---------------

3               Muzzle velocity (m/s). Refernce: BRL
1.0             Drag / Adjustment factor for MPI error
1.0             Aerodynamic lift of a projectile (%)
0.028125        Barrel elevation accuracy. Based on 0.5 mils accuracy. (deg)
4               Barrel azimuth accuracy. Based on 4 mils accuracy. (mils)
15              Gun location - range (m)
15              Gun location - deflection (m)
0               Chart accuracy - range (m)
0               Chart accuracy - deflection (m)
```

Figure 31. Sample Input file

Since there is an option to choose between a basic trajectory calculation or accuracy calculation, the output file is different. Figure 32 shows an output file when the program is selected to run as a basic trajectory calculation. Figure 33 shows the output file when the accuracy results were calculated.

```
---------PRECISION AND MPI ERRORS ERRORS CALCULATION---------


---------------Inputs for trajectory--------------------------

Projectile Mass: 42.000 kg                      Projectile Diameter: 0.155 m
Deflection Accln: 0.203 m/s^2                   Muzzle Velocity: 684.500 m/s
Target Range: 5000.000 m                        Target Direction: 0.000 deg
Wind direction: 0.000 deg                       Wind Speed:0.000 m/s
Temperature Variability: 1.000                  Density Variability: 1.000


---------------Nominal Trajectory Results--------------------

Range: 5031.077 m                               Barrel Elevation: 72.970 mils
Time of Flight: 9.200 s                         Maximum Altitude: 103.996 m
Cross wind correction: 0.000 mils               Drift correction: -1.738 mils
Total Deflection Correction: -1.738 mils
```

Figure 32. Sample output file showing trajectory results only

```
---------PRECISION AND MPI ERRORS ERRORS CALCULATION---------


---------------Inputs for trajectory--------------------------

Projectile Mass: 42.000 kg                      Projectile Diameter: 0.155 m
Deflection Accln: 0.203 m/s^2                   Muzzle Velocity: 684.500 m/s
Target Range: 15000.000 m                       Target Direction: 0.000 deg
Wind direction: 0.000 deg                       Wind Speed:0.000 m/s
Temperature Variability: 1.000                  Density Variability: 1.000


---------------Nominal Trajectory Results--------------------

Range: 15005.886 m                              Barrel Elevation: 1119.075 mils
Time of Flight: 85.120 s                        Maximum Altitude: 8999.433 m
Cross wind correction: 0.000 mils               Drift correction: -49.895 mils
Total Deflection Correction: -49.895 mils



---------------Input Std Deviations for Precision Error------

Muzzle Velocity: 1.990 m/s                      Drag(Adjusment): 0.650 %
Barrel Elevation: 0.017 deg                     Delfection constant a_0: 0.520
Deflection constant a_1: 2000.000

---------------Input Std Deviations for MPI Error------------

Wind Speed: 2.521 m/s                           Density: 0.690 %
Temp:0.570 %                                    Muzzle velocity: 3.000 m/s
Drag(Adjustment): 1.000 %                       Lift(Adjustment): 1.000 %
Elevation accuracy: 0.028 deg                   Azimuth accuracy: 4.000 deg
Location accuracy (X): 15.000 m                 Location accuracy(Z): 15.000 m
Chart accuracy (X): 0.000 m                     Chart Accuracy(Z): 0.000 m



---------------Partial Derivatives---------------------------

d(Range)/d(Elevation): -337.840                 d(Range)/d(Muzzle velocity): 25.013
d(Range)/d(Density): -78.361                    d(Range)/d(Temp): 11.426
d(Range)/d(Wind): 31.220
d(Deflection)/d(Drift): 7.350                   d(Deflection)/d(Azimuth): 14.732
d(Deflection)/d(Wind): 27.187


---------------Precision Errors------------------------------

               Range: 71.445 m                  Deflection: 25.785 m
```

Figure 33. Sample Output File

D. PROGRAM CODES – MAIN PROGRAM

```
%%%This program calculates the PE and MPI error for Predicted Fire------

format long;
clear all
clear global
clc

disp(' ');
disp('Analysis in progress');

%%%Specifying global parameters-----------------------------------------
global G R KAPPA CP CV DT D PROJECTILEMASS ACCLN_DEFLECTION
%%%End------------------------------------------------------------------

%%%Defining all constants-----------------------------------------------
%Atmospheric constants
G = 9.81;          %Gravitational Constant in meters/sec^2
R = 287.05;        %Universal Gas Constant
KAPPA = 1.402;     %Adiabatic index
CP = 1004.5;       %Specific heat of air (J/KgK @ 300K)
CV = 717.5;        %Specific heat of air (J/KgK @ 300K)

%Calculation constants
DT = 0.01;         %Define time step for trajectory calculation
%%%End of Constants-----------------------------------------------------

%%%Loading user inputs--------------------------------------------------
fid = fopen('Input.txt', 'r');
non = fscanf(fid, '%s',10);

non = fscanf(fid, '%s',2);
Error_cal = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 11);

non = fscanf(fid, '%s', 4);
PROJECTILEMASS = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 4);
D = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 5);
ACCLN_DEFLECTION = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 3);

non = fscanf(fid, '%s', 8);
x_desired = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 3);
Vt = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 3);
Vm_direction = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 7);
Elevation_flag = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 10);

non = fscanf(fid, '%s', 5);
Wind_direction = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 13);
Wind_speed = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 3);
Temp_variation = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 10);
Density_variation = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 10);
staleness_hour = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 14);

non = fscanf(fid, '%s', 7);
sigma_Vt_prec = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 5);
sigma_drag_prec = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 7);
```

```
sigma_theta_prec = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 8);
a_0  = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 5);
a_1 = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 5);

non = fscanf(fid, '%s', 7);
sigma_Vt_MPI = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 5);
sigma_DRAG_MPI = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 7);
sigma_LIFT_MPI = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 6);
sigma_AIM_EL_MPI = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 9);
sigma_AIM_AZ_MPI = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 9);
sigma_LOC_X_MPI = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 5);
sigma_LOC_Z_MPI = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 5);
sigma_CHART_X_MPI = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 5);
sigma_CHART_Z_MPI = fscanf(fid, '%f', 1);
non = fscanf(fid, '%s', 5);

fclose(fid);
%%%End----------------------------------------------------------------------

%%%Units Conversion
Wind_speed = Wind_speed * 0.514; %Conversion from knots to m/s
%%%End----------------------------------------------------------------------

%%%Variables/Initialisation--------------------------------------------------
x_range =0.05 * x_desired; %To predict the range to 5% of desired range
vary_temp = 0;             %Initialise variation in temp to 0
vary_rho  = 0;             %Initialise variation in density to 0
vary_windspeed_x = 0;   %Initialise variation in wind speed to 0 (Range)
vary_windspeed_z = 0;   %Initialise variation in wind speed to 0 (Dfln)
%%%End of defining variables-------------------------------------------------


%%%Calculating the minimum and maximum range given the launch velocity--
Vt_try = Vt;
theta_try = 0.1;        %Setting the minimum QE to 0.1 deg
[time, x, y, thetat, Vt1, r, h, Z_wind, Z_wind_array, Deflection_angle_corr,
Drift_deflection_corr, TDC] = Trajectory(Vt_try, theta_try, vary_rho, vary_temp,
vary_windspeed_x, vary_windspeed_z, Vm_direction, Wind_direction, Wind_speed,
Temp_variation, Density_variation);
x1 = x;

theta_try = 45;         %Setting the maximum WE to 45 deg
[time, x, y, thetat, Vt1, r, h, Z_wind, Z_wind_array, Deflection_angle_corr,
Drift_deflection_corr, TDC] = Trajectory(Vt_try, theta_try, vary_rho, vary_temp,
vary_windspeed_x, vary_windspeed_z, Vm_direction, Wind_direction, Wind_speed,
Temp_variation, Density_variation);
x2 = x;
%%%End of calculating-------------------------------------------------------

%%%Calculating the required launch angle-------------------------------------
if(x_desired>x1 & x_desired<x2) %If desired range is betw min & max range

    theta_1 = asind(x_desired*G/Vt^2)/2; %Est of QE at low elevation w/o drag

    theta_2 = 90 - theta_1;              %Est of QE at high elevation w/o drag

    %%%To calculate required QE-----------------------------------------
    if (Elevation_flag == 0)    %If lower elevation is preferred
    theta0 = theta_1;           %Initialise launch angle to zero
    x = 0;                      %Initialise range to zero
```

```
    while(x < x_desired - x_range)  %Rough calculation of launch elevation (0.5
deg steps)

    theta0 = theta0 + 0.5 ;
    [time, x, y, thetat, Vt1, r, h, Z_wind, Z_wind_array, Deflection_angle_corr,
Drift_deflection_corr, TDC] = Trajectory(Vt, theta0, vary_rho, vary_temp,
vary_windspeed_x, vary_windspeed_z, Vm_direction, Wind_direction, Wind_speed,
Temp_variation, Density_variation);                  %Calculating the trajectory
    end

    while(x < x_desired)      %Fine cal of launch elevation (0.05 deg)
    theta0 = theta0 + 0.05;
    [time, x, y, thetat, Vt1, r, h, Z_wind, Z_wind_array, Deflection_angle_corr,
Drift_deflection_corr, TDC] = Trajectory(Vt, theta0, vary_rho, vary_temp,
vary_windspeed_x, vary_windspeed_z, Vm_direction, Wind_direction, Wind_speed,
Temp_variation, Density_variation);
    end

    theta0_1 = theta0;         %Lower elevation launch angle

    elseif (Elevation_flag == 1) %If higher elevation is desired
    theta0 = theta_2;            %Initialise launch angle to 90
    x = 0;                       %Initialise range to zero

    while(x < x_desired - x_range)  %Rough calculation of launch elevation (0.5
deg steps)

    theta0 = theta0 - 0.5;
    [time, x, y, thetat, Vt1, r, h, Z_wind, Z_wind_array, Deflection_angle_corr,
Drift_deflection_corr, TDC] = Trajectory(Vt, theta0, vary_rho, vary_temp,
vary_windspeed_x, vary_windspeed_z, Vm_direction, Wind_direction, Wind_speed,
Temp_variation, Density_variation);   %Calculating the trajectory
    end

    while(x < x_desired)            %Fine calculation of launch elevation (0.05
deg)
    theta0 = theta0 - 0.05 ;
    [time, x, y, thetat, Vt1, r, h, Z_wind, Z_wind_array, Deflection_angle_corr,
Drift_deflection_corr, TDC] = Trajectory(Vt, theta0, vary_rho, vary_temp,
vary_windspeed_x, vary_windspeed_z, Vm_direction, Wind_direction, Wind_speed,
Temp_variation, Density_variation);   %Calculating the trajectory
    end
    theta0_2 = theta0;                  %Higher elevation launch angle

    end
    %%%End of calculation for higher elevation launch angle------------


    %%%Nominal Trajectory-----------------------------------------------

    if (Elevation_flag == 0)         %If lower elevation preferred
        theta0 = theta0_1;
    elseif(Elevation_flag == 1)      %If higher elevation is preferred
        theta0 = theta0_2;
    end

    [time, x, y, thetat, Vt1, r, h, Z_wind, Z_wind_array, Deflection_angle_corr,
Drift_deflection_corr, TDC] = Trajectory(Vt, theta0, vary_rho, vary_temp,
vary_windspeed_x, vary_windspeed_z, Vm_direction, Wind_direction, Wind_speed,
Temp_variation, Density_variation);   %Calculating the nominal trajectory

    Launch_angle = theta0*17.777778; %Conversion from degrees to mils
    Range = x;                       %Final Range
    Final_alt = y;                   %Final impact altitude
    Impact_angle = thetat*180/pi;    %Impact angle
    Impact_velocity = Vt1;           %Impact velocity
    Time_flight = time;              %Time of flight
    Height = max(h);                 %Maximum height
    Deflection_wind = Z_wind;        %Deflection due to wind
    Crosswind_correction = Deflection_angle_corr;   %Correction for deflection
due to wind
    Drift_correction = Drift_deflection_corr;       %Correction for drift
```

```
    Total_deflection_corr = TDC;     %Total correction for both drift &
deflection

    if(Error_cal==0);         %Print output file for trajectory only

    fid = fopen('Output.txt', 'wt');
    fprintf(fid, '\n');
    fprintf(fid, '---------PRECISION AND MPI ERRORS ERRORS CALCULATION---------
\n\n\n');

    fprintf(fid, '---------------Inputs for trajectory-------------------------
\n\n');
    fprintf(fid, 'Projectile Mass: %2.3f kg\t\t\tProjectile Diameter: %2.3f
m\n', PROJECTILEMASS, D);
    fprintf(fid, 'Deflection Accln: %2.3f m/s^2\t\t\tMuzzle Velocity: %2.3f
m/s\n', ACCLN_DEFLECTION, Vt);
    fprintf(fid, 'Target Range: %2.3f m\t\t\tTarget Direction: %2.3f deg\n',
x_desired, Vm_direction);
    fprintf(fid, 'Wind direction: %2.3f deg\t\t\tWind Speed:%2.3f m/s\n',
Wind_direction, Wind_speed);
    fprintf(fid, 'Temperature Variability: %2.3f\t\t\tDensity Variability:
%2.3f\n', Temp_variation, Density_variation);
    fprintf(fid, '\n\n');

    fprintf(fid, '---------------Nominal Trajectory Results-------------------
\n\n');
    fprintf(fid, 'Range: %2.3f m\t\t\t\tBarrel Elevation: %2.3f mils\n', Range,
Launch_angle);
    fprintf(fid, 'Time of Flight: %2.3f s\t\t\tMaximum Altitude: %2.3f m\n',
Time_flight, Height);
    fprintf(fid, 'Cross wind correction: %2.3f mils\t\tDrift correction: %2.3f
mils\n', Crosswind_correction,     Drift_correction);
    fprintf(fid, 'Total Deflection Correction: %2.3f mils\n',
Total_deflection_corr);
    fprintf(fid, '\n\n\n');

    end

    %%%End of Basic Trajectory Program---------------------------------

    %%%Calculation for PE and MPI error--------------------------------
    if(Error_cal==1);   %Cal the error margins and the partial derivatives

    [sigma_wind, sigma_density, sigma_temp] = Met(staleness_hour);
    sigma_rho_MPI = sigma_density; %Error budget for density with staleness hour
    sigma_WIND_MPI = sigma_wind;    %Error budget for wind speed with staleness
hour
    sigma_TEMP_MPI = sigma_temp;    %Error budget for temperature with staleness
hour

    %%%Defining variables for partial derivatives----------------------
    vary_theta = 10/17.7777;   %QE unit effect variation - 10 mils
    vary_Vt = 10;              %MV unit effect variation - 10 m/s
    %%%End of deifnition of variables for partial derivatives----------

    %%%Calculating Launch Angle Partial Derivative---------------------
    theta1=theta0 - 0*vary_theta;   %QE variation at 0 mils
    [time, x, y, thetat, Vt1, r1, h, Z_wind, Z_wind_array,
Deflection_angle_corr, Drift_deflection_corr, TDC] = Trajectory(Vt, theta1,
vary_rho, vary_temp, vary_windspeed_x, vary_windspeed_z, Vm_direction,
Wind_direction, Wind_speed, Temp_variation, Density_variation);
    x1=x;

    theta2=theta0 + vary_theta;     %QE variation at +10 mils
    [time, x, y, thetat, Vt1, r2, h, Z_wind, Z_wind_array,
Deflection_angle_corr, Drift_deflection_corr, TDC] = Trajectory(Vt, theta2,
vary_rho, vary_temp, vary_windspeed_x, vary_windspeed_z, Vm_direction,
Wind_direction, Wind_speed, Temp_variation, Density_variation);
    x2=x;

    dR_dtheta = (x2-x1)/(vary_theta);   %Partial derivatives in m/degree
    %%%End of calculation for the Launch Angle Partial Derivative------
```

```
    %%%Calculating Launch Velocity Partial Derivative-----------------
    Vt01 = Vt - 0*vary_Vt;          %MV variation at 0 m/s
    [time, x, y, thetat, Vt1, r1, h, Z_wind, Z_wind_array,
Deflection_angle_corr, Drift_deflection_corr, TDC] = Trajectory(Vt01, theta0,
vary_rho, vary_temp, vary_windspeed_x, vary_windspeed_z, Vm_direction,
Wind_direction, Wind_speed, Temp_variation, Density_variation);
    x1=x;

    Vt02 = Vt + vary_Vt;            %MV variation at +10m/s
    [time, x, y, thetat, Vt1, r2, h, Z_wind, Z_wind_array,
Deflection_angle_corr, Drift_deflection_corr, TDC] = Trajectory(Vt02, theta0,
vary_rho, vary_temp, vary_windspeed_x, vary_windspeed_z, Vm_direction,
Wind_direction, Wind_speed, Temp_variation, Density_variation);
    x2=x;

    dR_dVt = (x2-x1)/(vary_Vt);     %Partial derivatives in seconds
    %%%End of calculation for the Launch Velocity Partial Derivative---

    %%%Calculating Density Partial Derivative--------------------------
    vary_rho=1;        %Density variation at 0%
    [time, x, y, thetat, Vt1, r1, h, Z_wind, Z_wind_array,
Deflection_angle_corr, Drift_deflection_corr, TDC] = Trajectory(Vt, theta0,
vary_rho, vary_temp, vary_windspeed_x, vary_windspeed_z, Vm_direction,
Wind_direction, Wind_speed, Temp_variation, Density_variation);
    x1=x;

    vary_rho=2;                     %Density variation at +10%
    [time, x, y, thetat, Vt1, r2, h, Z_wind, Z_wind_array,
Deflection_angle_corr, Drift_deflection_corr, TDC] = Trajectory(Vt, theta0,
vary_rho, vary_temp, vary_windspeed_x, vary_windspeed_z, Vm_direction,
Wind_direction, Wind_speed, Temp_variation, Density_variation);
    x2=x;

    dR_drho = (x2-x1)/(10);        %Partial Derivatives in m/%

    vary_rho = 0;                   %Reset density to nominal condition
    %%%End of calculation for the Density Partial Derivative-----------

    %%%Calculating Temperature Partial Derivative----------------------
    vary_temp=1;                    %Temperature variation at 0%
    [time, x, y, thetat, Vt1, r1, h, Z_wind, Z_wind_array,
Deflection_angle_corr, Drift_deflection_corr, TDC] = Trajectory(Vt, theta0,
vary_rho, vary_temp, vary_windspeed_x, vary_windspeed_z, Vm_direction,
Wind_direction, Wind_speed, Temp_variation, Density_variation);
    x1=x;

    vary_temp=2;                    %Temperature variation by +10%
    [time, x, y, thetat, Vt1, r2, h, Z_wind, Z_wind_array,
Deflection_angle_corr, Drift_deflection_corr, TDC] = Trajectory(Vt, theta0,
vary_rho, vary_temp, vary_windspeed_x, vary_windspeed_z, Vm_direction,
Wind_direction, Wind_speed, Temp_variation, Density_variation);
    x2=x;

    dR_dtemp = (x2-x1)/(10) ;   %Partial dervivatives in m/%

    vary_temp = 0;              %Reset temperature to nominal condition
    %%%End of calculation for the Temperature Partial Derivative-------

    %%%Calculating Drift Partial Derivative----------------------------
    Z_deflection = 0.5*ACCLN_DEFLECTION*Time_flight^2;
    Drift = Z_deflection * 1018.59  / Range;         %Drift is in rads
    dZ_dDrift = (Drift / 100) * (Range / 1018.59);   %Partial derivatives in m/%
    %%%End of calculation for Drift Partial Derivatives----------------

    %%%Calculating Aiming in Azimuth Partial Derivative----------------
    dZ_dalpha = Range/1018.59;     %Partial derivatives in meters/mils
    %%%End of calculation----------------------------------------------

    %%%Calculating Wind speed Partial Derivatives for range direction--
    vary_windspeed_x = 1;          %Wind speed variation at 0 kts
```

```
    [time, x, y, thetat, Vt1, r1, h, Z_wind, Z_wind_array,
Deflection_angle_corr, Drift_deflection_corr, TDC] = Trajectory(Vt, theta0,
vary_rho, vary_temp, vary_windspeed_x, vary_windspeed_z, Vm_direction,
Wind_direction, Wind_speed, Temp_variation, Density_variation); %Calculate
trajectory due to + wind speed
    x1=x;

    vary_windspeed_x = 2;          %Wind speed variation at +10 kts
    [time, x, y, thetat, Vt1, r2, h, Z_wind, Z_wind_array,
Deflection_angle_corr, Drift_deflection_corr, TDC] = Trajectory(Vt, theta0,
vary_rho, vary_temp, vary_windspeed_x, vary_windspeed_z, Vm_direction,
Wind_direction, Wind_speed, Temp_variation, Density_variation);
    x2=x;

    dR_dwind = (x2-x1)/(10*0.5144);%Partial derivatives in 1/s

    vary_windspeed_x = 0;       %Reset wind speed to nominal condition
    %%%End----------------------------------------------------------------

    %%%Calculating Wind speed Partial Derivatives for range direction--
    vary_windspeed_z = 1;       %Wind speed variation at 0 kts
    [time, x, y, thetat, Vt1, r1, h, Z_wind, Z_wind_array,
Deflection_angle_corr, Drift_deflection_corr, TDC] = Trajectory(Vt, theta0,
vary_rho, vary_temp, vary_windspeed_x, vary_windspeed_z, Vm_direction,
Wind_direction, Wind_speed, Temp_variation, Density_variation); %Calculate
trajectory due to + wind speed
    z1=Z_wind;

    vary_windspeed_z = 2;      %Wind speed variation at +10 kts
    [time, x, y, thetat, Vt1, r2, h, Z_wind, Z_wind_array,
Deflection_angle_corr, Drift_deflection_corr, TDC] = Trajectory(Vt, theta0,
vary_rho, vary_temp, vary_windspeed_x, vary_windspeed_z, Vm_direction,
Wind_direction, Wind_speed, Temp_variation, Density_variation);
    z2=Z_wind;

    dZ_dwind = (z2-z1)/(10*0.5144);   %Partial derivatives in 1/s

    vary_windspeed_z = 0;   %Reset the wind speed to nominal condition
%%%End----------------------------------------------------------------

%%%Precision Error Model-----------------------------------------------
sigma_PX_PE =
sqrt((dR_dVt*sigma_Vt_prec)^2+(dR_drho*sigma_drag_prec)^2+(dR_dtheta*sigma_theta
_prec)^2);   %Std deviation of miss distance in range

sigma_PZ_PE = (a_0/1018.59)*((a_1*Range)/(a_1-Launch_angle))/0.6745;
%%%End of Precision Error----------------------------------------------

%%%MPI Error Model-----------------------------------------------------
sigma_X_MPI = sqrt( (dR_drho)^2*(sigma_rho_MPI^2+sigma_DRAG_MPI^2) +
(dR_dtemp*sigma_TEMP_MPI)^2 + (dR_dwind*sigma_WIND_MPI)^2 +
(dR_dVt*sigma_Vt_MPI)^2 + (dR_dtheta*sigma_AIM_EL_MPI)^2 +sigma_LOC_X_MPI^2 +
sigma_CHART_X_MPI^2 );  %MPI error in range

sigma_Z_MPI = sqrt( (dZ_dwind*sigma_WIND_MPI)^2 + (dZ_dDrift*sigma_LIFT_MPI)^2 +
(dZ_dalpha*sigma_AIM_AZ_MPI)^2 + sigma_LOC_Z_MPI^2 + sigma_CHART_Z_MPI^2 );
%MPI error in deflection
%%%END-----------------------------------------------------------------

    end

else

    disp(' ');
    disp('Invalid Range / Velocity Combination');
    x1 = num2str(x1);
    x1 = ['The minimum range is ', x1];
    disp(x1);
    x2 = num2str(x2);
    x2 = ['The maximum range is ', x2];
    disp(x2);
```

```
end

%%%Output results--------------------------------------------------------
if(Error_cal == 1)  %Print output file for both trajectory and error cal

fid = fopen('Output.txt', 'wt');
fprintf(fid, '\n');
fprintf(fid, '---------PRECISION AND MPI ERRORS ERRORS CALCULATION---------
\n\n\n');

fprintf(fid, '---------------Inputs for trajectory-------------------------
\n\n');
fprintf(fid, 'Projectile Mass: %2.3f kg\t\t\tProjectile Diameter: %2.3f m\n',
PROJECTILEMASS, D);
fprintf(fid, 'Deflection Accln: %2.3f m/s^2\t\t\tMuzzle Velocity: %2.3f m/s\n',
ACCLN_DEFLECTION, Vt);
fprintf(fid, 'Target Range: %2.3f m\t\t\tTarget Direction: %2.3f deg\n',
x_desired, Vm_direction);
fprintf(fid, 'Wind direction: %2.3f deg\t\t\tWind Speed:%2.3f m/s\n',
Wind_direction, Wind_speed);
fprintf(fid, 'Temperature Variability: %2.3f\t\t\tDensity Variability: %2.3f\n',
Temp_variation, Density_variation);
fprintf(fid, '\n\n');

fprintf(fid, '---------------Nominal Trajectory Results--------------------
\n\n');
fprintf(fid, 'Range: %2.3f m\t\t\t\tBarrel Elevation: %2.3f mils\n', Range,
Launch_angle);
fprintf(fid, 'Time of Flight: %2.3f s\t\t\tMaximum Altitude: %2.3f m\n',
Time_flight, Height);
fprintf(fid, 'Cross wind correction: %2.3f mils\t\tDrift correction: %2.3f
mils\n', Crosswind_correction, Drift_correction);
fprintf(fid, 'Total Deflection Correction: %2.3f mils\n',
Total_deflection_corr);
fprintf(fid, '\n\n\n');

fprintf(fid, '---------------Input Std Deviations for Precision Error------
\n\n');
fprintf(fid, 'Muzzle Velocity: %2.3f m/s\t\t\tDrag(Adjusment): %2.3f %%\n',
sigma_Vt_prec, sigma_drag_prec);
fprintf(fid, 'Barrel Elevation: %2.3f deg\t\t\tDelfection constant a_0:
%2.3f\n',  sigma_theta_prec, a_0');
fprintf(fid, 'Deflection constant a_1: %2.3f\n', a_1);
fprintf(fid, '\n');

fprintf(fid, '---------------Input Std Deviations for MPI Error------------
\n\n');
fprintf(fid, 'Wind Speed: %2.3f m/s\t\t\t\tDensity: %2.3f %%\n', sigma_WIND_MPI,
sigma_rho_MPI);
fprintf(fid, 'Temp:%2.3f %%\t\t\t\tMuzzle velocity: %2.3f m/s\n',
sigma_TEMP_MPI, sigma_Vt_MPI);
fprintf(fid, 'Drag(Adjustment): %2.3f %%\t\t\tLift(Adjustment): %2.3f %%\n',
sigma_DRAG_MPI, sigma_LIFT_MPI);
fprintf(fid, 'Elevation accuracy: %2.3f deg\t\t\tAzimuth accuracy: %2.3f deg\n',
sigma_AIM_EL_MPI, sigma_AIM_AZ_MPI);
fprintf(fid, 'Location accuracy (X): %2.3f m\t\t\tLocation accuracy(Z): %2.3f
m\n', sigma_LOC_X_MPI, sigma_LOC_Z_MPI);
fprintf(fid, 'Chart accuracy (X): %2.3f m\t\t\tChart Accuracy(Z): %2.3f m\n',
sigma_CHART_X_MPI, sigma_CHART_Z_MPI);
fprintf(fid, '\n\n\n');

fprintf(fid, '---------------Partial Derivatives--------------------------
\n\n');
fprintf(fid, 'd(Range)/d(Elevation): %2.3f\t\t\td(Range)/d(Muzzle velocity):
%2.3f\n', dR_dtheta, dR_dVt);
fprintf(fid, 'd(Range)/d(Density): %2.3f\t\t\td(Range)/d(Temp): %2.3f\n',
dR_drho, dR_dtemp);
fprintf(fid, 'd(Range)/d(Wind): %2.3f\n', dR_dwind);
fprintf(fid, 'd(Deflection)/d(Drift): %2.3f\t\t\td(Deflection)/d(Azimuth):
%2.3f\n', dZ_dDrift, dZ_dalpha);
fprintf(fid, 'd(Deflection)/d(Wind): %2.3f\n', dZ_dwind);
fprintf(fid, '\n\n');
```

```
fprintf(fid, '---------------Precision Errors-----------------------------
\n\n');
fprintf(fid, '\t\tRange: %2.3f m\t\t\tDeflection: %2.3f m\n\n', sigma_PX_PE,
sigma_PZ_PE);

fprintf(fid, '---------------MPI Errors-----------------------------------
\n\n');
fprintf(fid, '\t\tRange: %2.3f m\t\tDeflection: %2.3f m\n\n', sigma_X_MPI,
sigma_Z_MPI);
fclose(fid);

end
%End----------------------------------------------------------------------

disp('Analysis Completed - See Output.txt');
 %%%End of Program-------------------------------------------------------
```

E. PROGRAM CODES – TRAJECTORY M-FILE

```
function [time, x, y, thetat, Vt1, r, h, Z_wind, Z_wind_array,
Deflection_angle_corr, Drift_deflection_corr, TDC] = Trajectory(Vt, theta0,
vary_rho, vary_temp, vary_windspeed_x, vary_windspeed_z, Vm_direction,
Wind_direction, Wind_speed, Temp_variation, Density_variation)

global G R KAPPA CP CV DT D PROJECTILEMASS ACCLN_DEFLECTION

%%%Trajectory calculation program-------------------------------------

theta0 = theta0*pi/180;      %Conversion from deg to radians
Vx = Vt*cos(theta0);         %Horizontal component of projectile's velocity
Vy = Vt*sin(theta0);         %Vertical component of projectile's velocity


%%%Initialisation-----------------------------------------------------
ACCLN_DEFLECTION = 0.2029; %Accleration due to drift
x = 0;                      %Initialise x to zero
y = 0;                      %Initialise y to zero
n = 0;                      %Initialise number of cycles to zero
thetat = theta0;            %Initialise flight angle to launch angle for intial
conditions
Vt1=Vt;                     %Initialise projectile's velocity to MV
V_Z_wind = 0;              %Initialise the deflection speed due to wind to zero
Z_wind = 0;               %Initialise the deflection distance due to wind to zero
%%%End----------------------------------------------------------------

%%%Main calculation---------------------------------------------------
while(y>=0)      %Calculate as long as the altitude is more than zero
n = n+1;         %Number of loops

[temp,      rho]=Accoustic(y,      vary_rho,      vary_temp,      Temp_variation,
Density_variation); %Density and temperature according to altitude

Vs = sqrt(KAPPA*R*temp);     %Calculate speed of sound
Vw_direction = (Wind_direction - Vm_direction)*pi/180; %Calculate angle of wind
to the range direction in horizontal plane

Wind_speed_X = Wind_speed*cos(Vw_direction);     %Range component of the wind
speed
Wind_speed_Z = Wind_speed*sin(Vw_direction);      %Deflection component of the
wind speed
Wind_speed_New = Wind_speed;                     %Re-assign name to wind speed

if (vary_windspeed_x == 1);                      %Only for calculating range wind
unit effect
    Wind_speed_X = Wind_speed_X - 0*0.5144;       %Variation of wind speed in
range by 0 kts
    Wind_speed_New = sqrt(Wind_speed_X^2+Wind_speed_Z^2); %To calculate the new
wind speed
```

```
     if (Wind_speed_New == 0);
        Vw_direction = (Wind_direction - Vm_direction)*pi/180;
     else
     Vw_direction = acos(Wind_speed_X/Wind_speed_New);     %To calculate the new
wind direction
     end

    elseif (vary_windspeed_x == 2);             %Only for calculating range wind
unit effect
    Wind_speed_X = Wind_speed_X + 10*0.5144;      %Variation of wind speed in
deflection by +10 kts
    Wind_speed_New = sqrt(Wind_speed_X^2+Wind_speed_Z^2); %To calculate the new
wind speed
    Vw_direction = acos(Wind_speed_X/Wind_speed_New);      %To calculate the new
wind direction
end

if (vary_windspeed_z == 1);                        %Only for calculating deflectn
wind unit effect
    Wind_speed_Z = Wind_speed_Z - 0*0.5144;        %Variation of wind speed in
deflectn by 0 kts
    Wind_speed_New = sqrt(Wind_speed_X^2+Wind_speed_Z^2);  %To calculate the new
wind speed
    if (Wind_speed_New == 0);
        Vw_direction = (Wind_direction - Vm_direction)*pi/180;
    else
    Vw_direction = asin(Wind_speed_Z/Wind_speed_New);      %To calculate the new
wind direction
    end

elseif (vary_windspeed_z == 2);                    %Only for calculating deflectn
wind unit effect
    Wind_speed_Z = Wind_speed_Z + 10*0.5144;       %Variation of wind speed in
deflectn by +10 kts
    Wind_speed_New = sqrt(Wind_speed_X^2+Wind_speed_Z^2);  %To calculate the new
wind speed
    Vw_direction = asin(Wind_speed_Z/Wind_speed_New);      %To calculate the new
wind direction
end

Wind_speed_proj = -Wind_speed_New*cos(thetat);    %Resolve the wind speed in the
projectile axis plane

theta_WP = pi - Vw_direction;                      %Calculate the angle between
the wind and trajectory

V_RW = sqrt(Vt1^2 + Wind_speed_proj^2 - 2*Vt1*Wind_speed_proj*cos(theta_WP));
%Calculate the absolute relative projectile speed
theta_RW = asin(Wind_speed_proj*sin(theta_WP)/V_RW);    %Calculate the angle of
the relative projectile speed

M = V_RW / Vs;                             %Calculate Mach number for the relative
wind speed
Cd = Drag_coeff(M);                       %Function file for drag coefficient
Fd = 0.5*0.25*pi*D*D*rho*V_RW^2*Cd;    %Drag force in the projectile relative
speed direction

Fd_X = Fd * cos(theta_RW);            %Resolve drag force in range direction
Fd_Z = Fd * sin(theta_RW);                  %Resolve drag force in deflection
direction

%%%Trajectory for down range-----------------------------------------
ax = -Fd_X*cos(thetat)/PROJECTILEMASS;    %Compute the x acceleration
ay = -G-Fd_X*sin(thetat)/PROJECTILEMASS;  %Compute the y acceleration

Vx1 = ax*DT+Vx;                           %Compute x velocity in t+dt
Vy1 = ay*DT+Vy;                           %Compute y velocity in t+dt

x1 = Vx1*DT+x;                            %Compute x displacement in t+dt
y1 = Vy1*DT+y;                            %Compute y displacement in t+dt
%%%-----------------------------------------------------------------
```

```
%%%Trajectory for deflection due to wind------------------------------
if (Vw_direction >= 0 & Vw_direction <= 180);
    Fd_Z = Fd_Z;

else
    Fd_Z = -Fd_Z;

end

a_Z_wind = -Fd_Z/PROJECTILEMASS;   %Calculate the acceleration in the deflection
direction due to wind
V_Z_wind = a_Z_wind*DT + V_Z_wind; %Calculate the velocity of the projectile in
the deflection direction due to wind
Z_wind = V_Z_wind*DT + Z_wind;     %Calculate deflection distance due to wind
%%%-------------------------------------------------------------------
x = x1;                 %Initialise x for the next loop
y = y1;                 %Initialise y for the next loop
Vx=Vx1;                 %Initialise Vx for the next loop
Vy=Vy1;                 %Initialise Vy for the next loop
thetat=atan2(Vy,Vx);    %Initalise flight angle for the next loop
Vt1=sqrt(Vx^2+Vy^2);    %Initialise the flight velocity for the next loop

h(n)=y;                 %Array for y for n values - Alitude
r(n)=x;                 %Array for x for n values - Range
Z_wind_array(n)=Z_wind;%Array for z for n values - Deflection

end

%%% Output Results of Trajectory Program-------------------------------
time = n*DT;                        %Total time of flight
x;                                  %Output Range
y;                                  %Output final altitude
thetat;                             %Output Impact angle
Vt1;                                %Output Impact velocity
Z_wind;                             %Output delfection due to wind

Deflection_angle_corr = - Z_wind * 1018.59 / x;  %Deflection angle correction in
mils

%%%Calculate the deflection due to drift-----------------------------
Drift_deflection = 0.5*ACCLN_DEFLECTION*time^2;
Drift_deflection_corr = -Drift_deflection*1018.59/x;  %Drift is in mils
%%%End---------------------------------------------------------------


%%%Total drift correction---------------------------------------------
TDC = Deflection_angle_corr + Drift_deflection_corr;
%%%End---------------------------------------------------------------
```

F. PROGRAM CODES – ACOUSTIC M-FILE

```
%%%Function file for acoustic----------------------------------------
%%%IACO standard atmospheric data

function  [temp,  rho]  =  Accoustic(y,  vary_rho,  vary_temp,  Temp_variation,
Density_variation)

z       = [0        1000      1999      2999      3997      4996      5994      6992
7990      8987      9984      10981     11977     12973     13969     14965     15960
16955     17949     18943     19937     20931];        %height
t       = [288.150  281.651  275.154  268.659  262.166  255.676  249.187  242.700
236.215   229.733   223.252   216.774   216.650   216.650   216.650   216.650   216.650
216.650   216.650   216.650   216.650   217.581];      %temp in K
r       = [1.2250    1.1117    1.0066    0.90925  0.81935  0.73643  0.66011  0.59002
0.52579   0.46706   0.41351   0.3648    0.31194  0.2666    0.22786  0.19475  0.16647
0.14230   0.12165   0.10400   0.088910 0.075715];
%air density

i=1;
while(z(i)<=y)
```

```
    i=i+1;
end

rho      = (( y - z(i-1) ) / ( z(i)-z(i-1) ) * ( r(i) - r(i-1) ) + r(i-
1))*Density_variation; %Calculation for air density for required altitude
temp     = (( y - z(i-1) ) / ( z(i)-z(i-1) ) * ( t(i) - t(i-1) ) + t(i-
1))*Temp_variation;    %Calculation for air temperature for required altitude

if(vary_rho==1)             %Only for unit effect calculation
      rho = rho*1;          %Density variation at 0%
    elseif(vary_rho==2)
  rho = rho*1.1;            %Density variation by +10%
 end

if(vary_temp==1)            %Only for unit effect calculation
   temp = temp*1;           %Temperature variation at 0%
elseif(vary_temp==2)
   temp = temp*1.1;         %Temperature variation by +10%
end

%%%End----------------------------------------------------------------
```

G. PROGRAM CODES – MET M-FILE

```
function [sigma_wind, sigma_density, sigma_temp] = Met(staleness_hour)

z         = [0    1    2    4    5];      %Staleness hour
s_wind    = [0.8  4.0  4.9  7.2  11.0 ];  %Standard deviation in knots
s_density = [0.15 0.40 0.69 0.97 6.60];   %Standard deviation of density
s_temp    = [0.25 0.30 0.57 0.79 3.00];   %standard deviation of temp

i=1;
while(z(i)< staleness_hour)
    i=i+1;
end

sigma_wind = s_wind(i)*0.5144444444;      %Conversion to m/s
sigma_density = s_density(i);
sigma_temp = s_temp(i);
```

H. PROGRAM CODES – DRAG_COEFF M-FILE

```
%%%Drag coefficient at particular Mach---------------------------------
function Cd=Drag_coeff(x)

M =  [0     0.2   0.5   0.7   0.8   0.85 0.9   0.92  0.96 1.02  1.04  1.07  1.1
1.25  1.5   2.0   2.5  3.0  4.0];  %Mach numner
Cd =  [0.121  0.121  0.121  0.121  0.122  0.13  0.152  0.198  0.27  0.335  0.382  0.389
0.391  0.371  0.325  0.294  0.26  0.26  0.26]; %Drag Coefficient at particular Mach

i=1;
while M(i)<x
i=i+1;
end

Cd=(x-M(i-1))/(M(i)-M(i-1))*(Cd(i)-Cd(i-1))+Cd(i-1);     %Interpolating     Drag
Coefficient
```

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX II. DRAG COEFFICIENT AND DRAG CURVE

The aerodynamic drag coefficient, C_D, is a combination of the wave drag, friction drag and base drag and the drag force can be calculated using equation (2.2). The drag coefficient is a function of the Mach number which can varies with altitude. The aerodynamic drag coefficient used in this report is as shown in the Table 13. The plot of the drag coefficient against the Mach number can be seen in Figure 34. The software program performs a linear interpolation of the drag coefficient for Mach numbers that are not in the table such as 0.95 Mach.

Mach	0	0.2	0.5	0.7	0.8	0.85	0.9	0.92	0.96
C_D	0.121	0.121	0.121	0.121	0.122	0.13	0.152	0.197	0.27
Mach	1.02	1.04	1.07	1.1	1.25	1.5	2.0	2.5	3.0
C_D	0.335	0.382	0.389	0.391	0.371	0.325	0.294	0.26	0.26

Table 13. Aerodynamic Drag Coefficient

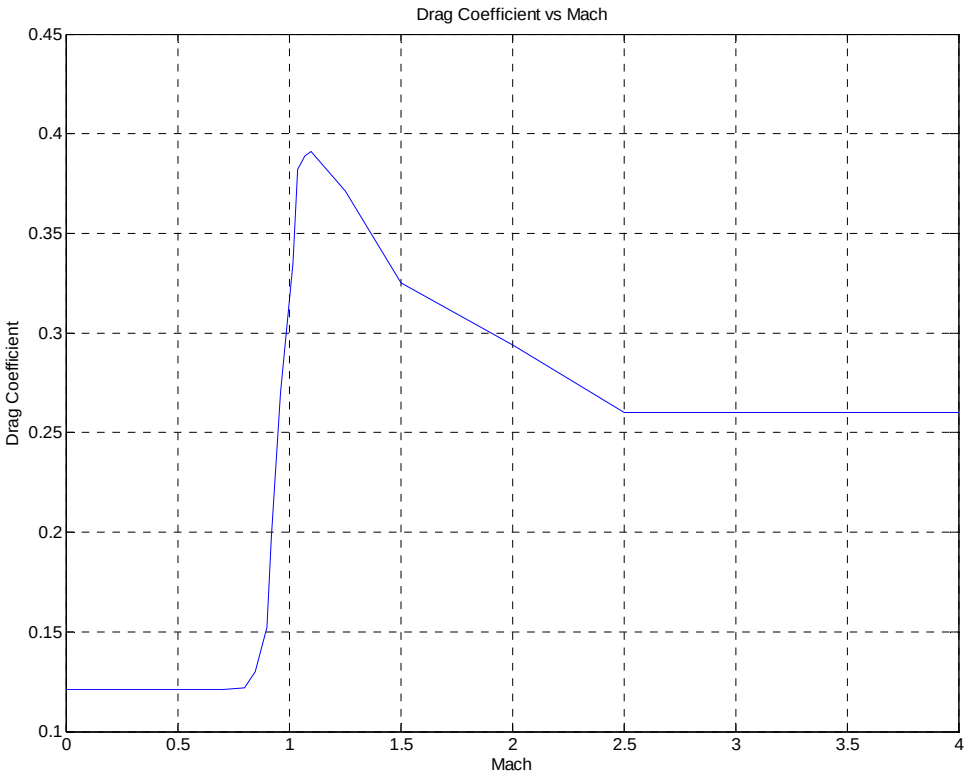

Figure 34. Plot of Drag Coefficient with Mach

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX III. SAMPLE CALCULATION OF THE DEFLECTION ACCELERATION

In Chapter III, the various ways of determining the drift of a projectile was discussed. The NABK is of 5 DOF and thus able to predict the projectile's drift accurately. On the other hand, due to the complexity of the equations, a simpler way of calculating the drift has to be formulated. This was discussed in Chapter III and it involves the determination of the projectile's acceleration in the deflection direction. With the acceleration known, the drift due to the projectile can be found using equation (3.23).

$$Z_p = \frac{1}{2} a_{D,p} t^2 . \qquad \text{(A3.1)}$$

The $a_{D,p}$ is the estimated cumulative lateral acceleration of the projectile due to the gyroscopic effect determined from actual data for a particular projectile. The drift and time of flight is known from existing data, therefore from equation (3.9) and using existing data,

$$a_{D,p} = \frac{2Z_p}{t^2} . \qquad \text{(A3.2)}$$

A sample data is as shown in Table 14, which has ranges from 0 to 4, 000 meters. Using equation (D.1), the acceleration due to the projectile can be found as shown in the last column. The acceleration is averaged out between a range of 0 and 14, 500 meters to increase its accuracy. It was found that the mean of the acceleration is 0.203 meters per seconds square and a standard deviation of only 0.04 meters per second square. The behavior of the acceleration with range is plotted in Figure 35.

Range	Elevation	Time of Flight (s)	Drift (mils)	Drift (rad)	Drift (meters)	Drift Acceleration (m/s^2)
100	1.8	0.2	0	0.0000	0.0000	0.0000
200	3.5	0.4	0.1	0.0001	0.0196	0.2454
300	5.2	0.6	0.1	0.0001	0.0295	0.1636
400	6.9	0.7	0.2	0.0002	0.0785	0.3206
500	8.7	0.9	0.2	0.0002	0.0982	0.2424
600	10.4	1.1	0.3	0.0003	0.1767	0.2921
700	12.2	1.3	0.3	0.0003	0.2062	0.2440
800	14	1.5	0.3	0.0003	0.2356	0.2094
900	15.8	1.7	0.4	0.0004	0.3534	0.2446
1000	17.6	1.9	0.4	0.0004	0.3927	0.2176
1100	19.4	2.1	0.5	0.0005	0.5400	0.2449
1200	21.2	2.3	0.5	0.0005	0.5890	0.2227
1300	23.2	2.5	0.6	0.0006	0.7658	0.2450
1400	25.1	2.7	0.6	0.0006	0.8247	0.2262
1500	27	2.9	0.6	0.0006	0.8836	0.2101
1600	28.9	3.1	0.7	0.0007	1.0996	0.2288
1700	30.9	3.3	0.7	0.0007	1.1683	0.2146
1800	32.9	3.5	0.8	0.0008	1.4137	0.2308
1900	35	3.7	0.8	0.0008	1.4923	0.2180
2000	37	3.9	0.9	0.0009	1.7671	0.2324
2100	39.1	4.1	0.9	0.0009	1.8555	0.2208
2200	41.2	4.4	1	0.0010	2.1598	0.2231
2300	43.3	4.6	1	0.0010	2.2580	0.2134
2400	45.5	4.8	1.1	0.0011	2.5918	0.2250
2500	47.7	5	1.1	0.0011	2.6998	0.2160
2600	49.9	5.3	1.2	0.0012	3.0631	0.2181
2700	52.2	5.5	1.2	0.0012	3.1809	0.2103
2800	54.5	5.7	1.3	0.0013	3.5736	0.2200
2900	56.8	6	1.4	0.0014	3.9859	0.2214
3000	59.1	6.2	1.4	0.0014	4.1233	0.2145
3100	61.5	6.4	1.5	0.0015	4.5651	0.2229
3200	64	6.7	1.5	0.0015	4.7124	0.2100
3300	66.4	6.9	1.6	0.0016	5.1836	0.2178
3400	68.9	7.2	1.6	0.0016	5.3407	0.2060
3500	71.4	7.4	1.7	0.0017	5.8414	0.2133
3600	74	7.7	1.8	0.0018	6.3617	0.2146
3700	76.6	7.9	1.8	0.0018	6.5385	0.2095
3800	79.2	8.2	1.9	0.0019	7.0882	0.2108
3900	81.9	8.5	1.9	0.0019	7.2748	0.2014
4000	84.6	8.7	2	0.0020	7.8540	0.2075

Table 14. Sample data on the calculation of the acceleration in drift

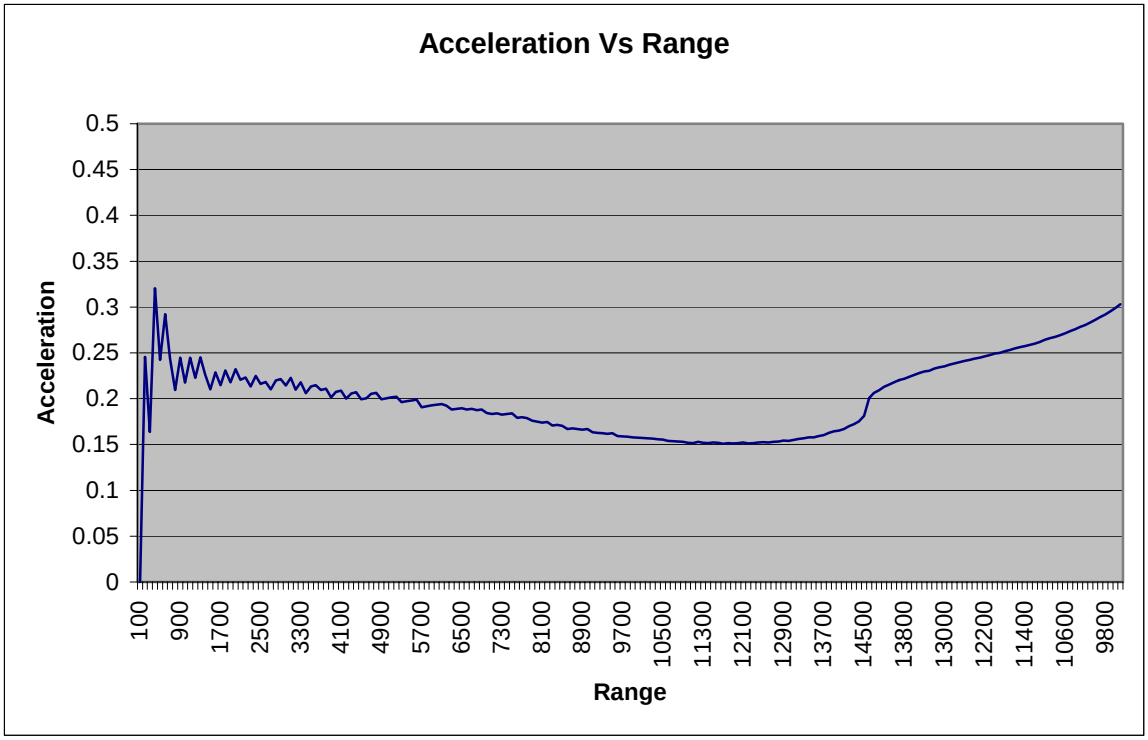

Figure 35. Plot of Projectile's Drift Acceleration with Range

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

1. Driels, Morris R., "Weaponeering: Conventional Weapon System Effectiveness," Reston, Virginia, American Institute of Aeronautics and Astronautics, 2004.

2. Backman, Marvin E., "Terminal Ballistics," China Lake, California, Naval Weapons Center, 1976.

3. Moulton, Forest R., "Methods in Exterior Ballistics," New York, Dover Publication, 1962.

4. Farrar, C. L. and D. W. Leeming, "Military Ballistics – A Basic Manual," Oxford, Brassey's Publishers Limited.

5. Matts, James A., "Physically Consistent Delivery Accuracy Models for Cannon Artillery," Aberdeen Proving Ground, Maryland, U.S. Army Armament Research, Development and Engineering Center, 1995.

6. Matts, James A. and Andrew G. Ellis, "Artillery Accuracy: Simple Models to Assess The Impact of New Equipment and Tactics," Technical report BRL-TR-3101, Aberdeen Proving Ground, Maryland, Ballistics Research Laboratory, 1990.

7. NATO STANAG 4355 Modified Point Mass Trajectory Model

8. Bray, Derek. "External Ballistics." Cranfield University, Shrivenham, accessed December 1, 2006, http://www.dcmt.cranfield.ac.uk/aeroxtra/exballs.htm.

9. Ruprecht, Nennstiel. "How Do Bullets Fly" Wiesbaden, Germany, accessed November 30, 2006, http://www.nennstiel-ruprecht.de/bullfly.

10. Federation of American Scientist, Washington, D.C., accessed October 25, 2006, http://www.fas.org/man/dod-101/sys/index.html.

11. FM6-40/MCWP, "Tactics, Techniques, and Procedures for Field Artillery Manual Cannon Gunnery," April 1996

12. International Civil Aviation Organization, Standard Atmosphere Manual, Washington, D.C.: GPO, 1964.

13. FT 155-AM-2, "Firing Tables for Cannon 155 mm Howitzer:, Headquarters, Department of the Army, 31 March 1983.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Mr. J. Matts
   U.S. Army ARDEC
   Attn: AMSTA-AR-FSF-T
   Aberdeen Proving Ground, Maryland

4. Mr. T. Hills
   Analysis Division, U.S. Army Field Artillery School
   Attn: ATSF-FCA
   Fort Sill, Oklahoma

5. Ms. S. Hughes
   NSWC Dahlgren Division
   Dahlgren, Virginia

6. Professpr Yeo Tat Soon
   Director, Temasek Defence Systems Institute (TDSI)
   National University of Singapore
   Singapore